REPMARK: REPRESENTATION-STABLE RADIOACTIVE LATENT WATERMARKING FOR DIFFUSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Watermarks for image provenance do not necessarily survive downstream model training, limiting their use for tracing unauthorized training data. We introduce RepMark, a radioactive latent watermark designed to remain recoverable across representation changes and to propagate into diffusion models. RepMark maps a multibit owner message to a bounded displacement in shared writing coordinates, creating a consistent signal across protected images. A LoRA-adapted decoder writes the signal, while a private LoRA-adapted encoder and a proxy-VAE training path support message extraction after representation changes. The resulting interface supports post-processing of completed images, in-processing through compatible VAE replacement, and pre-processing for training-data tracing. In the last setting, the watermark is inherited through training, eliminating an additional online writing step at deployment and enabling black-box verification from novel model outputs. Across PixArt fine-tuning and LightningDiT training from scratch, RepMark achieves 0.999–1.000 per-image presence AUROC in the evaluated settings. Multi-image aggregation further recovers the registered owner message, including after protected training data are encoded by an external SD3.5 VAE, while maintaining image quality and limited separability under specified public-representation tests.

1 INTRODUCTION

Visual generative models have substantially lowered the cost of producing high-quality images, but they also create new challenges for content provenance and digital asset protection. Digital watermarking embeds an owner-associated, verifiable signal in an image and provides a practical means of tracing image provenance and training data use (Fernandez et al., 2023; Sablayrolles et al., 2020). The latter requires *radioactivity*: after protected training images are used to train or fine-tune a student model, evidence associated with the owner should remain statistically detectable in novel images generated by that model (Zhao et al., 2024; Kerner et al., 2025).

Existing image watermarks are usually deployed at one of two positions. *Post-processing* methods protect a completed image through a transform domain or learned embedder. They are largely independent of the source model, but subsequent image transformations may weaken their signals (Zhu et al., 2018; Tancik et al., 2020; Zhao et al., 2024). *In-processing* methods embed signals inside the generation pipeline, for example in diffusion noise, a VAE decoder, or model parameters (Wen et al., 2023; Fernandez et al., 2023; Kim et al., 2023). They can reliably mark outputs of a controlled generator, but are tied to its architecture. Neither path directly addresses *pre-processing*, where images must be protected before they enter an unknown training pipeline and the resulting radioactive evidence must later be recovered from a new model's outputs. Indeed, prior evaluations show that source-model detectability does not guarantee inheritance through downstream diffusion training (Dubiński et al., 2025).

Pre-processing for radioactive tracing faces two coupled challenges. First, VAE compression, diffusion noise, and parameter optimization can attenuate a watermark before it reaches the representation learned by the student. Second, if the writing direction changes with image content, watermark updates can cancel across the training set. The signal should therefore remain recoverable across heterogeneous representations and recur coherently across samples. At the same time, it should avoid conspicuous changes in the public representations examined by an external observer.

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.

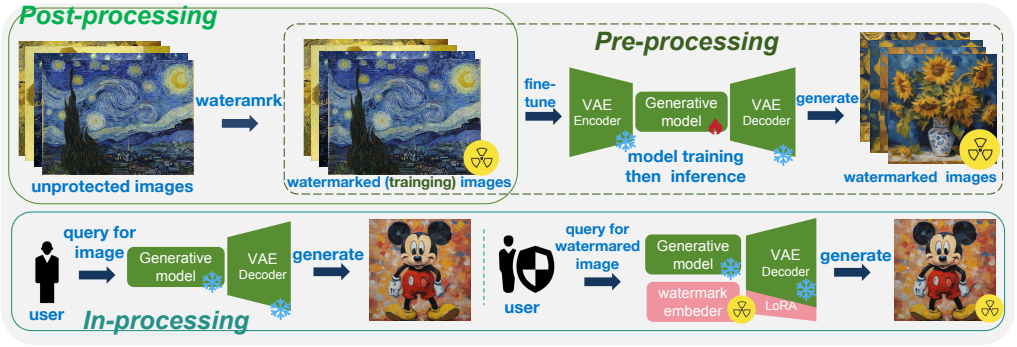


Figure 1: The unified RepMark system and its three deployment paths. Post-processing protects completed images. In-processing embeds a watermark through a latent-compatible VAE interface. Pre-processing protects training data and verifies inherited watermark evidence in novel outputs of the downstream model.

Motivated by these challenges, we propose **RepMark**, a radioactive latent watermark that distills reliable model-level tracing into two complementary design objectives: stable recovery across heterogeneous representations and consistent writing across training samples. To promote cross-sample consistency, a frozen VAE establishes writing coordinates shared across carriers, where an L-bit owner message produces a bounded displacement independent of image content. A LoRA-adapted decoder writes this displacement into pixel space. To promote cross-representation stability, a private LoRA-adapted encoder maps observations into an owner-controlled verification space, while a frozen proxy-VAE path explicitly trains presence detection and message extraction after representation changes. Image-quality constraints limit visible distortion, while public-space separability, private recoverability, and final model-level attribution are evaluated separately.

As shown in Figure 1, RepMark supports post-processing of completed images, in-processing through compatible VAE replacement, and pre-processing before model training. The pre-processing path verifies inherited evidence from novel outputs and moves watermark insertion offline: an owner-trained model can produce statistically marked outputs without invoking the writer during deployment, while the same mechanism enables black-box auditing of third-party models from queried outputs.

Our contributions are threefold. First, we characterize reliable radioactive propagation through two complementary mechanisms: cross-sample-consistent writing, which supports accumulation across protected data, and cross-representation recovery, which preserves verifiable evidence through heterogeneous encoding paths. We distinguish both from final model-level attribution on novel outputs. Second, we introduce RepMark, a multibit latent watermark that writes carrier-independent displacements in frozen shared VAE coordinates and uses bilateral LoRA with direct and proxy-VAE paths for private verification. Third, we systematically evaluate radioactivity under model fine-tuning and training from scratch across resolutions, training budgets, protected-data fractions, and an unseen VAE, and validate post-processing, in-processing, and pre-processing deployments.

2 Background and Problem Formulation

2.1 Related Work

Image watermarking includes post-processing methods that protect completed images and in-processing methods that embed signals inside a controlled generator (Zhu et al., 2018; Fernandez et al., 2023). Both primarily provide image-level verification; successful detection in this setting does not by itself establish inheritance by a separately trained model (Dubiński et al., 2025).

Training-data tracing instead studies evidence propagated from protected images into downstream model outputs. Existing methods either modify released data with repeated or optimized signals, or create marked images through internal representations and sampling procedures of a source generator (Cui et al., 2023; Li et al., 2025a;b; Kociszewski et al., 2026). They typically specialize in one writing interface rather than the full image–generation–training chain.

RepMark uses one shared VAE interface for post-processing, in-processing, and radioactive training-data protection. It combines carrier-independent multibit writing, private verification, and proxy-VAE recovery, while evaluating cross-representation recovery, cross-sample consistency, and final model-level propagation separately. Detailed comparisons are deferred to Appendix A.

2.2 Latent Generative Models

Latent generative models map images between pixel space $\mathcal{X}$ and latent space $\mathcal{Z}$ using encoder $\mathcal{E}$ and decoder $\mathcal{D}$, then learn the latent distribution through noising and denoising (Rombach et al., 2022). A protected image is therefore affected by VAE reconstruction, diffusion noise, and optimization before its evidence can reach a downstream model. Moreover, different systems may use VAEs with different architectures, channel counts, downsampling rates, and objectives, giving the same image different latent dimensions and coordinate meanings. Both surviving these representation changes and being subsequently learned by the downstream model are challenges that a radioactive watermark must overcome.

2.3 Radioactive Watermarking: Problem Formulation

Radioactive watermarking spans writing, data processing, downstream training, and owner verification. We use *representation space* for the pixel, frequency, codec-latent, and learned-feature spaces encountered along this chain. Let $\{\mathcal{Z}_v\}_{v\in\mathcal{V}}$ denote these spaces and Φ_v their maps, which need not be VAEs or have public inverses. An owner holds an L-bit message $b \in \{0,1\}^L$ and protects image x as $x^w = \mathrm{Emb}(x, b)$. Its clean and watermarked representations are

$$z_v = \Phi_v(x), \qquad z_v^w = \Phi_v(x^w). \tag{1}$$

A data user may train generator G_θ with an unknown VAE, architecture, and procedure. The owner has black-box access only. For held-out prompt and seed (c_q, ξ_q), let $y_q = G_\theta(c_q, \xi_q)$. The owner queries Q outputs and aggregates private verification evidence:

$$S(G_\theta) = \mathrm{Agg}(\mathrm{Ver}(y_1), \ldots, \mathrm{Ver}(y_Q)) \,. \tag{2}$$

Ver returns a presence score or multibit prediction, and Agg is the associated multi-image rule. Evidence exceeding a threshold calibrated with clean references at the same query budget indicates that protected training data left a detectable trace in the downstream model's output distribution. The owner neither observes model parameters nor intervenes in training.

2.4 Evaluation Objectives

We separate three properties, illustrated in Figure 2. The left panel characterizes the relations among carrier content, watermark signal, and representation noise in the shared writing space $\mathcal{Z}_1$, a proxy representation space $\mathcal{Z}_2$, and the private verification space $\mathcal{Z}_3$. The right panel uses t-SNE to visualize how clean and watermarked samples are distributed after encoding by different VAEs, showing their local mixing or separation in each representation space.

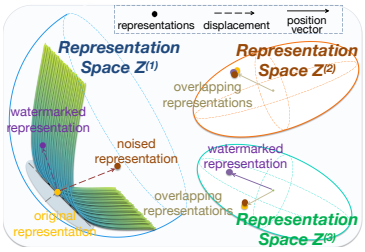
The first property is cross-representation stability. For transformations $\mathcal{T}_v$ through the evaluated representations, we seek

$$\Pr[\mathrm{Ver}(\mathcal{T}_v(x^w)) = b] \geq 1 - \epsilon_{\mathrm{rec}}, \qquad v \in \mathcal{V}_{\mathrm{eval}}. \tag{3}$$

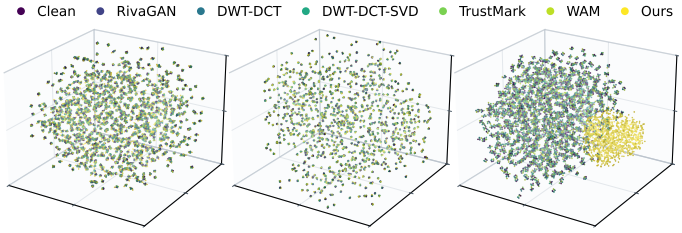
The watermark need not share dimensions or numerical form across spaces; it only needs to remain recoverable through the private interface. A complementary requirement is empirical concealment: for a prespecified discriminator family $\mathcal{F}_v$ in a public representation, we seek

$$\sup_{f\in\mathcal{F}_v} |\Pr[f(z_v^w) = 1] - \Pr[f(z_v) = 1]| \leq \epsilon_{\mathrm{ind}}. \tag{4}$$

This concerns only the selected public spaces and discriminator families rather than indistinguishability against an arbitrary adversary. Finally, model-level radioactivity requires the aggregate in Equation (2) to exceed a threshold calibrated on outputs from separately trained clean-reference students. Stability can preserve a signal without ensuring that the downstream model learns it, and concealment alone does not determine propagation, so final attribution relies on novel outputs from the model under investigation.



(a) Carrier, watermark, and noise across three spaces.



(b) t-SNE distribution of the writing representation, proxy representation, and verification representation space.

Figure 2: Cross-representation signal geometry and distributions. Left: relations among carrier content, watermark signal, and noise in the shared writing, proxy, and private verification spaces. Right: t-SNE distributions of clean and watermarked samples under different VAE representations.

3 Method

3.1 Design Motivation: Shared Writing Coordinates

Using the representation spaces defined in Section 2.3, a class of watermarking methods can be expressed as additive modulation in a representation domain. Let $\mathcal{E}$ be a representation map and $\mathcal{D}$ a reconstruction map:

$$z = \mathcal{E}(x), \qquad z^w = z + \delta_b, \qquad x^w = \mathcal{D}(z^w), \tag{5}$$

where δ_b is the representation displacement associated with message b, and $\mathcal{D}$ need not be the exact inverse of $\mathcal{E}$. Both classical transform-domain and learned watermarks can be written in this form, but they differ in how their writing coordinates are established.

RepMark's central design is to write messages in coordinates shared across samples. Under per-image SVD, for example, every carrier x_i has a different singular basis, so the perturbation direction can vary with image content even when the same message is embedded. A frozen VAE encoder instead uses the same network parameters to map every image into shared writing space $\mathcal{Z}_1$. For a fixed owner message $b^\star$, RepMark produces a carrier-independent message displacement $\delta_{b^\star}$, so

$$z_i^w = z_i + \delta_{b^\star}. \tag{6}$$

Although content representations z_i differ, $\delta_{b^\star}$ recurs in the same shared coordinates. We therefore expect watermark signals in different training samples to remain more directionally consistent and to accumulate more readily in the downstream model. Shared writing coordinates are a favorable condition for model-level radioactivity, not a sufficient guarantee; propagation must still be tested on novel images generated by the downstream model.

3.2 RepMark Overview

RepMark instantiates the writing and black-box verification procedure in Section 2.3. Given owner message b, it produces a bounded displacement δ_b in shared writing space $\mathcal{Z}_1$ and embeds it through an adapted VAE decoder $\mathcal{D}_\phi$. For verification, an observed image is mapped to $\mathcal{Z}_3$, where the owner's private verification interface recovers message evidence.

As shown in Figure 1, the same watermark interface supports three deployment paths. In post-processing, RepMark obtains a latent from an existing image using the frozen encoder $\mathcal{E}_0$, adds the message displacement, and generates the protected image with the adapted decoder:

$$x \xrightarrow{\mathcal{E}_0} z \xrightarrow{+\delta_b} z^w \xrightarrow{\mathcal{D}_\phi} x^w. \tag{7}$$

In in-processing, a generator with compatible latent parameterization instead adds the message displacement directly to the denoiser output z_{gen} and decodes it with $\mathcal{D}_\phi$:

$$z_{\text{gen}} \xrightarrow{+\delta_b} z_{\text{gen}}^w \xrightarrow{\mathcal{D}_\phi} x_{\text{gen}}^w, \tag{8}$$

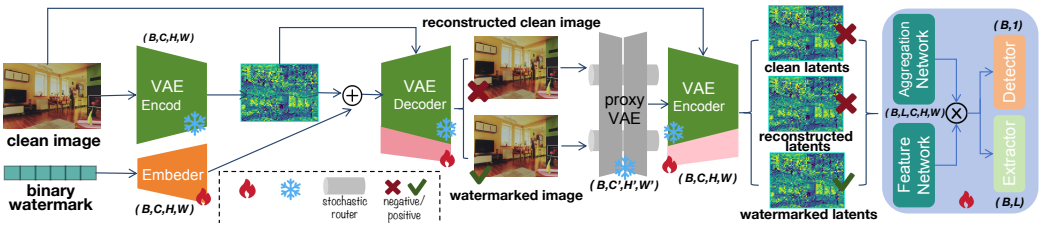


Figure 3: RepMark training. The frozen encoder $\mathcal{E}_0$ provides the writing latent, the writer maps the owner message to a bounded displacement, and the LoRA decoder $\mathcal{D}_\phi$ produces the watermarked image. Matched clean and watermarked samples then follow either the direct or frozen proxy-VAE path before entering the private verification space through the LoRA encoder $\mathcal{E}_\phi$ for message extraction and presence detection.

thereby avoiding the generation and re-encoding of a completed image. For pre-processing, the owner applies a fixed message $b^\star$ to construct the protected training set

$$\mathcal{D}^w = \{(x_i^w, c_i)\}_{i=1}^N. \tag{9}$$

After downstream model G_θ is trained on data containing $\mathcal{D}^w$, the owner queries novel outputs with held-out prompts and aggregates their evidence using Equation (2).

The first two paths support single-image presence detection and message extraction, whereas the third aggregates evidence across multiple outputs of the downstream model. All three share the same message representation, writer, and private verification interface.

3.3 Cross-Representation Writing and Recovery

RepMark instantiates the propagation chain in Section 2.3 with four spaces: pixel space $\mathcal{X}$, in which images propagate; shared writing space $\mathcal{Z}_1$, defined by the frozen base VAE; proxy representation space $\mathcal{Z}_2$, defined by the frozen SDXL VAE used during training; and private verification space $\mathcal{Z}_3$, adapted by the owner. The proxy space models a representation transformation that a data user might employ and does not assume that the proxy VAE matches the downstream model's VAE.

As shown in Figure 3, the frozen encoder $\mathcal{E}_0$ supplies the shared writing coordinates. Its LoRA-adapted decoder writes the latent displacement into pixel space as $x^w = \mathcal{D}_\phi(\mathcal{E}_0(x) + \delta_b)$, allowing the signal to propagate with the image. For verification, the LoRA-adapted encoder $\mathcal{E}_\phi$ maps an observation into $\mathcal{Z}_3$. The adapted decoder and encoder form the writing and verification channels without being required to constitute an analytic inverse pair.

For each watermarked image x^w, the matched clean negative is either the original image x or the no-displacement reconstruction $x' = \mathcal{D}_\phi(\mathcal{E}_0(x))$. Positive and negative samples follow the same route, preventing the verifier from relying only on reconstruction artifacts. To promote cross-representation recovery, a stochastic router sends each matched clean–watermarked pair through either the frozen proxy-VAE path or the direct path. Samples from both paths jointly supervise private message extraction and presence detection.

3.4 Message Writing and Private Verification

Bilateral LoRA adapts the writing and verification channels. The message writer $\mathcal{W}_\psi$ maps the L-bit owner message b to a bounded displacement with the shape of the writing latent:

$$\delta_b = \epsilon \tanh(\mathcal{W}_\psi(b; h, w)), \qquad \|\delta_b\|_\infty \le \epsilon, \tag{10}$$

where (h, w) is the latent resolution and ϵ is a learnable amplitude. Because the writer depends only on the message and target resolution, a fixed message induces the same displacement across carriers.

The private verification interface comprises the LoRA encoder $\mathcal{E}_\phi$ and message extraction and presence detection modules that share low-level evidence. For an observed image y, the private encoder produces $\widetilde{z} = \mathcal{E}_\phi(y)$. Feature and attention networks output bit-wise maps $F(\widetilde{z})$ and $A(\widetilde{z})$. After normalizing attention over channel–position support Ω, the extractor computes

$$e_i(y) = \sum_{p \in \Omega} A_{i,p}(\widetilde{z}) F_{i,p}(\widetilde{z}), \qquad \widehat{b}_i(y) = \mathbb{I}[e_i(y) > 0]. \tag{11}$$

The sign of $e_i(y)$ predicts bit i, while its magnitude measures evidence strength.

The presence detector reuses F and A but does not infer presence directly from the signs of $e_i(y)$. For each bit, it forms weighted evidence $E_{i,p} = A_{i,p}(\widetilde{z})F_{i,p}(\widetilde{z})$ and extracts a seven-dimensional descriptor $r_i(y)$ summarizing evidence magnitude, strength, and sign consistency over the full support and its strongest units. A shared per-bit judge with bit-specific scale and bias produces

$$u_i(y) = \alpha_i g_{\text{bit}}(r_i(y)) + \beta_i, \qquad i = 1, \ldots, L, \tag{12}$$

where $u_i(y)$ measures whether the ith evidence channel contains a structured watermark response, rather than whether its message bit is zero or one. The detector finally feeds the sorted per-bit presence evidence into global classifier $\mathcal{G}$ to obtain the image-level presence logit and score:

$$d(y) = \mathcal{G}(\text{sort}\,(u_1(y), \ldots, u_L(y))), \qquad s_{\text{det}}(y) = \sigma(d(y)), \tag{13}$$

where σ is the sigmoid function and $s_{\text{det}}(y) \in (0, 1)$ is the confidence that y contains a RepMark signal. Sorting makes the global classifier depend on how many evidence channels support presence, not on a fixed message-bit order. The extractor decodes the owner message from evidence directions, whereas the detector uses message-value-independent evidence strength and consistency to determine presence. They share evidence but use distinct reductions and supervision targets.

RepMark groups its objective into image-quality, message-extraction, and presence-detection terms:

$$\mathcal{L} = \mathcal{L}_{\text{qual}} + \lambda_{\text{extract}}\mathcal{L}_{\text{msg}} + \mathcal{L}_{\text{det}}. \tag{14}$$

The terms constrain visual and latent distortion, supervise message extraction on the direct and proxy-VAE paths, and distinguish matched watermarked and clean samples. Full sub-losses, weights, and sample construction are provided in Appendix B.

4 Experiments

4.1 Experimental Setup

We evaluate RepMark under model fine-tuning and training from scratch, analyze its cross-sample and cross-representation behavior, and compare its three deployment paths. Complete training configurations, evaluation protocols, and additional robustness and editing experiments are provided in Appendices B and C.

RepMark is trained on 5,000 generated images at resolutions 256^2, 512^2, and 1024^2, using prompts from Stable-Diffusion-Prompts. Messages are sampled independently during watermark training, whereas each radioactivity experiment uses a fixed owner message on an independent image–text dataset. Test prompts are disjoint from student-training captions. The base VAE remains frozen; only its encoder and decoder LoRA, the writer, and the private verification modules are trained. A frozen SDXL VAE provides the proxy cross-representation path. Unless stated otherwise, the message length is $L = 16$.

Verification comprises presence detection and message extraction. For each student output y, the presence detector produces $s_{\text{det}}(y)$. Per-image AUROC and TPR at 1% FPR compare these scores with outputs from separately trained clean-reference students and measure whether the signal appears in the protected student's output distribution; they do not identify the owner. Protected and clean-reference students use the same architecture, data, and training protocol, differing only in whether the training images are watermarked. Unless stated otherwise, model-level results are means over three independent student-training seeds.

For message extraction, we aggregate predictions from Q outputs of the same student by majority voting at each bit position and report their agreement with the registered message $b^\star$:

$$\widehat{b}_i^{G,Q} = \text{Majority}_{q=1}^{Q}\, \widehat{b}_i(y_q), \qquad \text{BitAcc}_Q = \frac{1}{L}\sum_{i=1}^{L} \mathbb{I}[\widehat{b}_i^{G,Q} = b_i^\star]. \tag{15}$$

We refer to this metric as query-aggregated BitAcc; outputs from different students are never combined in one decision. Exact recovery requires a positive presence decision and correct extraction of every message bit.

Table 1: Watermark presence in student outputs. Left: per-image AUROC after PixArt fine-tuning, evaluated using 500 outputs from each protected student and its separately trained clean-reference student. Right: per-image AUROC and TPR@1%FPR for LightningDiT trained from scratch, using 512 outputs per student. Values are means over three independent student-training seeds.

Method	PixArt fine-tuning AUROC			LightningDiT from scratch					
	256	512	1024	20K		40K		100K	
DiffusionShield	0.532	0.567	0.452	AUROC	TPR	AUROC	TPR	AUROC	TPR
CoprGuard	0.518	0.721	0.598						
DWT–DCT	0.508	0.448	0.512	0.545	1.6	0.597	5.5	0.614	3.9
DWT–DCT–SVD	0.509	0.529	0.533	0.561	18.2	0.606	21.7	0.590	24.4
RivaGAN	0.580	0.719	0.729	0.880	43.6	0.974	79.3	0.982	86.3
TrustMark-Q	0.504	0.471	0.511	0.989	94.1	0.999	99.4	1.000	99.8
Watermark Anything	0.536	0.715	0.777	0.989	97.3	1.000	100.0	1.000	100.0
RepMark	**0.999**	**1.000**	**1.000**	**0.999**	**99.6**	**1.000**	**100.0**	**1.000**	**100.0**

4.2 Watermark Propagation under Fine-Tuning and Training from Scratch

We evaluate watermark propagation under PixArt fine-tuning and LightningDiT training from scratch. PixArt is fine-tuned on fixed-message protected training images and evaluated using 500 held-out-prompt generations per method and resolution. LightningDiT is randomly initialized and trained under a shared dataset and configuration as a separate student for each method–budget setting, with 512 outputs evaluated per student. All AUROC and TPR values in Table 1 are computed separately for each training seed and then averaged over the three seeds; generations from one student are not treated as independent training runs.

Table 1 shows distinct propagation behavior across the two training regimes. After PixArt fine-tuning, RepMark achieves 0.999–1.000 AUROC at all three resolutions, whereas the evaluated baselines obtain 0.45–0.78. Under training from scratch, RepMark already reaches 0.999 AUROC and 99.6% TPR at 20K steps. TrustMark-Q and Watermark Anything also become strongly detectable as training proceeds, and RivaGAN improves more gradually, whereas the classical frequency-domain methods remain substantially weaker. Learned methods use a common trained encoder–decoder across images, providing a shared learned writing representation, whereas local transform-domain operations do not explicitly align their effective perturbations across carriers. This contrast is consistent with cross-sample writing consistency facilitating downstream accumulation; the controlled shared-coordinate ablation in Appendix C.5 tests this mechanism directly. Query-aggregated message extraction and complete results across training budgets are reported in Appendices C.1.

4.3 Representation Mechanism and Empirical Concealment

We examine three related but non-equivalent properties: whether a fixed message induces a cross-sample-consistent latent displacement, whether clean and watermarked images remain empirically mixed in specified public representations, and whether the watermark can cross the external SD3.5 representation, which is excluded from watermark training, and continue to propagate.

Directional coherence and local mixing. For 1,000 matched clean/watermarked pairs, we compute directional coherence on variance-standardized descriptor displacements $\Delta z_i = z_i^w - z_i$:

$$C = \frac{\left\| N^{-1} \sum_i \Delta z_i \right\|_2}{N^{-1} \sum_i \|\Delta z_i\|_2}. \tag{16}$$

Larger C indicates greater cross-sample directional consistency. Local mixing M_{200} is the fraction of opposite-label samples among the 200 nearest neighbors after joint standardization and PCA; values near 0.5 indicate local mixing. We evaluate the base and LoRA-adapted VAEs, the SDXL proxy VAE used in training, and the external SD3.5 VAE. All statistics use the high-dimensional descriptors rather than t-SNE coordinates; descriptor construction is given in Appendix B.

RepMark has the highest C in all four spaces, showing strongly aligned cross-sample displacements under this metric. Yet the proxy SDXL and external SD3.5 spaces remain locally mixed, with $M_{200} = 0.494$ and 0.472, respectively. The private verification space instead separates the watermark evidence by design ($M_{200} = 0.006$), while the shared writing space retains some separable

Table 2: Directional coherence C and local mixing M_{200} across four latent representations.

	Shared writing space $\mathcal{Z}_1$		Verification space $\mathcal{Z}_3$		SDXL proxy space $\mathcal{Z}_2$		External SD3.5 space	
Method	C	M_{200}	C	M_{200}	C	M_{200}	C	M_{200}
RivaGAN	0.564	0.474	0.544	0.464	0.151	0.500	0.738	0.490
DWT–DCT	0.426	0.491	0.473	0.447	0.277	0.499	0.642	0.495
DWT–DCT–SVD	0.487	0.488	0.524	0.469	0.415	0.499	0.660	0.496
TrustMark-Q	0.429	0.493	0.316	0.486	0.182	0.499	0.295	0.500
Watermark Anything	0.667	0.437	0.752	0.309	0.401	0.498	0.819	0.477
RepMark	**0.756**	0.362	**0.933**	0.006	**0.423**	0.494	**0.823**	0.472

Table 3: Image quality, public-representation statistics, and robustness at 256^2. Quality and representation metrics use 5,000 images. Pixel and log-spectrum descriptors report directional coherence C and local mixing M_{200}; $\text{AUROC}_{\text{avg}}$ averages presence AUROC over conventional post-processing attacks. RepMark reports same-latent incremental distortion, whereas post-processing methods use the original clean image as reference.

	Image quality				Pixel representation and robustness			Frequency representation	
Method	LPIPS↓	SSIM↑	CLIP↑	FID↓	C	M_{200}	$\text{AUROC}_{\text{avg}}$↑	C	M_{200}
RivaGAN	0.0136	0.9891	32.65	2.571	0.226	0.500	0.998	0.176	0.500
DWT–DCT	0.0062	0.9554	32.33	1.034	0.424	0.500	0.640	0.844	0.310
DWT–DCT–SVD	0.0106	0.9612	32.20	1.643	0.657	0.500	0.772	0.843	0.312
TrustMark-Q	0.0009	0.9854	32.51	0.862	0.700	0.500	1.000	0.925	0.453
Watermark Anything	0.0283	0.9613	32.43	4.309	0.224	0.500	1.000	0.686	0.493
DiffusionShield	0.0086	0.8553	32.02	3.047	0.588	0.500	0.766	0.964	0.410
CoprGuard	0.0008	0.9680	32.26	0.367	0.308	0.500	0.781	0.563	0.437
RepMark	0.0147	0.9721	32.84	1.075	0.673	0.499	0.955	0.918	0.461

structure ($M_{200} = 0.362$). Thus, the coherent signal persists across representation changes without strong local separation in the tested public spaces, but becomes separable in the owner's private space. This is empirical concealment under the specified tests, not general indistinguishability. Watermark Anything is also coherent but transfers poorly under PixArt fine-tuning, so coherence alone is insufficient for radioactivity. Linear-probe results are provided in Appendix C.2.1.

Image quality, public representations, and robustness. Table 3 evaluates image quality, pixel- and frequency-space behavior, and robustness at 256^2. For RepMark, we compare $\mathcal{D}_\phi(z)$ with $\mathcal{D}_\phi(z + \delta_b)$ from the same latent, isolating the incremental distortion caused by watermark writing. Post-processing baselines are compared with their original clean images, and all methods use their native operating points.

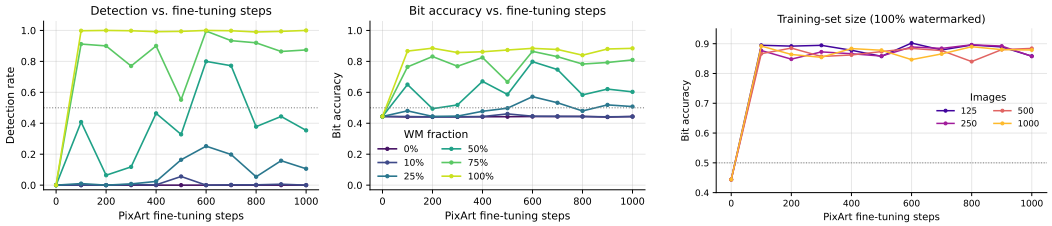
All methods have pixel-space M_{200} near 0.5, indicating substantial local mixing under the specified descriptor, but not general undetectability. RepMark achieves LPIPS 0.0147, SSIM 0.9721, and FID 1.075 under its same-latent control, together with 0.955 mean presence AUROC under conventional post-processing. Its frequency displacement is strongly coherent ($C = 0.918$) while remaining locally mixed ($M_{200} = 0.461$) under the evaluated metric. Thus, a cross-sample-consistent displacement need not produce strong local separation in the tested public pixel or frequency representations.

Protected fraction and external representations. Figure 4 studies the effects of protected-data fraction and training-set size under PixArt fine-tuning. Fractions up to 10% yield no stable transfer in this setting, while presence detection emerges between 25% and 50% and reaches 87.4% and 100% at 75% and 100%, respectively. The transition therefore depends on both the protected fraction and student-training setting.

Figure 5 evaluates propagation through the external SD3.5 representation under LightningDiT training from scratch. RepMark is trained with its writing VAE and an SDXL proxy VAE, whereas 500K protected training images are encoded by the external SD3.5 VAE for LightningDiT training. At 40K steps, the detection rate reaches 100% at both 256^2 and 512^2; at 80K steps, BitAcc reaches 0.9471 and 0.9651 without aggregation, showing that propagation is not confined to the proxy VAE.

Table 4: Three deployment paths. Native original-model outputs are clean controls; pre-processing uses native outputs of the model fine-tuned on protected data. All values are percentages.

		Original model			Fine-tuned model		
Resolution	Metric	Native	Post-processing	In-processing	Pre-processing	Post-processing stack	In-processing stack
256	Detection rate	0.1	100.0	100.0	99.9	100.0	100.0
	BitAcc	61.48	99.71	99.77	95.17	87.74	87.54
	Exact recovery	0.0	95.5	96.5	36.1	1.1	0.2
512	Detection rate	0.0	100.0	100.0	100.0	100.0	100.0
	BitAcc	59.15	99.98	100.00	98.71	87.66	87.56
	Exact recovery	0.0	99.8	100.0	80.8	0.4	0.2



(a) Detection rate and BitAcc across protected fractions.

(b) BitAcc versus training-set size.

Figure 4: Effects of protected-data fraction and training-set size under PixArt fine-tuning at 1024^2.

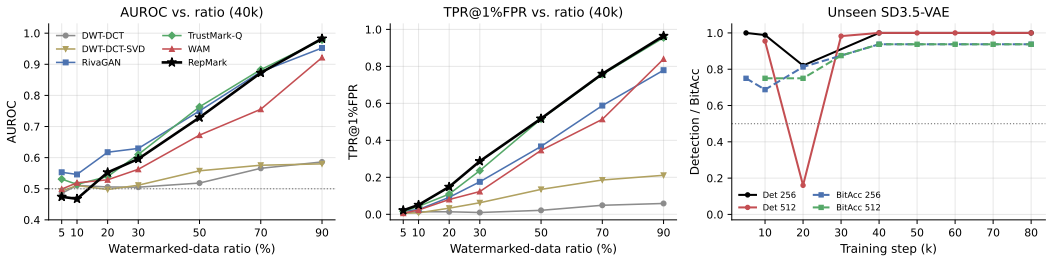


Figure 5: LightningDiT training from scratch: protected-fraction response (left, center) and propagation through an unseen SD3.5 VAE (right).

4.4 Three Deployment Paths

We compare the three paths using matched prompts, seeds, and denoised latents. Original-model outputs evaluate native, post-, and in-processing, while native outputs of a model fine-tuned on protected data evaluate pre-processing. Each group contains 5,000 images at 256^2 and 500 at 512^2.

Post- and in-processing achieve 100% detection and nearly perfect BitAcc. Pre-processing outputs reach 99.9–100% detection and 95.17–98.71% BitAcc, confirming inheritance without an online writer. Reapplying either writer preserves detection but lowers BitAcc to about 87.5% and nearly removes Exact recovery, showing that repeated writing introduces systematic bit errors. Latency, per-bit diagnostics, robustness, and qualitative examples are provided in Appendices C.3 and C.4.

5 Conclusion

We analyze how cross-sample writing consistency and cross-representation recovery support radioactive propagation. Based on these conditions, we develop RepMark, a full-lifecycle multibit watermarking system spanning post-processing, in-processing, and training-data protection. Through inheritance, RepMark supports writer-free online protection for owner-controlled models and black-box attribution of third-party models. Experiments on PixArt and LightningDiT, including an unseen SD3.5 VAE, validate propagation under fine-tuning and training from scratch while exposing its dependence on the protected-data fraction and training regime.

AI USE STATEMENT

AI tools were used for language polishing and to assist in preparing line plots for the image experiments. All experimental data, analyses, scientific claims, and substantive manuscript content were authored, checked, and verified by the authors, who take full responsibility for the paper.

REFERENCES

Shruti Agarwal, Tu Bui, and John Collomosse. TrustMark: Robust watermarking and watermark removal for arbitrary resolution images. In *IEEE/CVF International Conference on Computer Vision*, 2025.

Vishal Asnani, John Collomosse, Tu Bui, Xiaoming Liu, and Shruti Agarwal. ProMark: Proactive diffusion watermarking for causal attribution. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10802–10811, 2024.

Brian Chen and Gregory W. Wornell. Quantization index modulation: A class of provably good methods for digital watermarking and information embedding. *IEEE Transactions on Information Theory*, 47(4):1423–1443, 2001. doi: 10.1109/18.923725.

Chun-Hsien Chou and Yun-Chin Li. A perceptually tuned subband image coder based on the measure of just-noticeable-distortion profile. *IEEE Transactions on Circuits and Systems for Video Technology*, 5(6):467–476, 1995.

Ingemar Cox, Matthew Miller, Jeffrey Bloom, Jessica Fridrich, and Ton Kalker. *Digital Watermarking and Steganography*. Morgan Kaufmann, 2007.

Yingqian Cui, Jie Ren, Han Xu, Pengfei He, Hui Liu, Lichao Sun, and Jiliang Tang. DiffusionShield: A watermark for data copyright protection against generative diffusion models. *arXiv preprint arXiv:2306.04642*, 2023.

Yingqian Cui, Jie Ren, Yuping Lin, Han Xu, Pengfei He, Yue Xing, Wenqi Fan, Hui Liu, and Jiliang Tang. FT-Shield: A watermark against unauthorized fine-tuning in text-to-image diffusion models. *ACM SIGKDD Explorations Newsletter*, 26(2):76–88, 2024. doi: 10.1145/3715073.3715080.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Guang Shi, and Haoqi Fan. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

Luke Ditria and Tom Drummond. Hey that's mine! imperceptible watermarks are preserved in diffusion generated outputs. *arXiv preprint arXiv:2308.11123*, 2023.

Jan Dubiński, Michel Meintz, Franziska Boenisch, and Adam Dziedzic. Are watermarks for diffusion models radioactive? In *1st Workshop on GenAI Watermarking*, 2025. URL `https://openreview.net/forum?id=gtXbVRMwQh`.

Pierre Fernandez, Guillaume Couairon, Hervé Jégou, Matthijs Douze, and Teddy Furon. The stable signature: Rooting watermarks in latent diffusion models. In *IEEE/CVF International Conference on Computer Vision*, 2023.

Nir Goren, Oren Katzir, Abhinav Nakarmi, Eyal Ronen, Mahmood Sharif, and Or Patashnik. Noiseprints: Distortion-free watermarks for authorship in private diffusion models. In *The Fourteenth International Conference on Learning Representations*, 2026.

Xiaoxiao Hu, Jiaqi Jin, Sheng Li, Wanli Peng, Xinpeng Zhang, and Zhenxing Qian. Spherical watermark: Encryption-free, lossless watermarking for diffusion models. In *The Fourteenth International Conference on Learning Representations*, 2026.

Louis Kerner, Michel Meintz, Bihe Zhao, Franziska Boenisch, and Adam Dziedzic. BitMark: Watermarking bitwise autoregressive image generative models. In *Advances in Neural Information Processing Systems*, 2025.

Changhoon Kim, Kyle Min, Maitreya Patel, Sheng Cheng, and Yezhou Yang. WOUAF: Weight modulation for user attribution and fingerprinting in text-to-image diffusion models. *arXiv preprint arXiv:2306.04744*, 2023.

Jan Kociszewski, Hubert Jastrzebski, Tymoteusz Stepkowski, Filip Manijak, Krzysztof Rojek, Franziska Boenisch, and Adam Dziedzic. SERUM: Simple, efficient, robust, and unifying marking for diffusion-based image generation. In *The Fourteenth International Conference on Learning Representations*, 2026.

Boheng Li, Yanhao Wei, Yankai Fu, Zhenting Wang, Yiming Li, Jie Zhang, Run Wang, and Tianwei Zhang. Towards reliable verification of unauthorized data usage in personalized text-to-image diffusion models. In *IEEE Symposium on Security and Privacy*, pp. 2564–2582, 2025a. doi: 10.1109/SP61157.2025.00073.

Kexin Li, Guozhen Ding, Ilya Grishchenko, and David Lie. HMark: Radioactive multi-bit semantic-latent watermarking for diffusion models. *arXiv preprint arXiv:2512.00094*, 2025b.

Zhenguang Liu, Chao Shuai, Shaojing Fan, Ziping Dong, Jinwu Hu, Zhongjie Ba, and Kui Ren. Harnessing frequency spectrum insights for image copyright protection against diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18653–18662, 2025.

Xinyang Lu, Xinyuan Niu, Gregory Kang Ruey Lau, Nhung Bui, Rachael Hwee Ling Sim, John Russell Himawan, Fanyu Wen, Chuan-Sheng Foo, See-Kiong Ng, and Bryan Kian Hsiang Low. WaterDrum: Watermark-based data-centric unlearning metric. In *The Fourteenth International Conference on Learning Representations*, 2026.

Michel Meintz, Jan Dubiński, Franziska Boenisch, and Adam Dziedzic. Radioactive watermarks in diffusion and autoregressive image generative models. *arXiv preprint arXiv:2506.23731*, 2025.

Jie Ren, Yingqian Cui, Chen Chen, Vikash Sehwag, Yue Xing, Jiliang Tang, and Lingjuan Lyu. EnTruth: Enhancing the traceability of unauthorized dataset usage in text-to-image diffusion models with minimal and robust alterations. *arXiv preprint arXiv:2406.13933*, 2024.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022. doi: 10.1109/CVPR52688.2022.01042.

Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, and Hervé Jégou. Radioactive data: Tracing through training. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 8326–8335. PMLR, 2020.

Tom Sander, Pierre Fernandez, Alain Durmus, Teddy Furon, and Matthijs Douze. Watermark anything with localized messages. In *The Thirteenth International Conference on Learning Representations*, 2025.

ShieldMnt. `invisible-watermark`: Invisible watermarks for images. `https://github.com/ShieldMnt/invisible-watermark`, 2020.

Matthew Tancik, Ben Mildenhall, and Ren Ng. StegaStamp: Invisible hyperlinks in physical photographs. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2117–2126, 2020.

Xincheng Wang, Hanchi Sun, Wenjun Sun, Kejun Xue, Wangqiu Zhou, Jianbo Zhang, Wei Sun, Dandan Zhu, Xiongkuo Min, Jun Jia, and Zhijun Fang. Evaluating dataset watermarking for fine-tuning traceability of customized diffusion models: A comprehensive benchmark and removal approach. *arXiv preprint arXiv:2511.19316*, 2025.

Zhenting Wang, Chen Chen, Lingjuan Lyu, Dimitris N. Metaxas, and Shiqing Ma. DIAGNOSIS: Detecting unauthorized data usages in text-to-image diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024.

Yuxin Wen, John Kirchenbauer, Jonas Geiping, and Tom Goldstein. Tree-ring watermarks: Fingerprints for diffusion images that are invisible and robust. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

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

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. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Jiren Zhu, Russell Kaplan, Justin Johnson, and Li Fei-Fei. HiDDeN: Hiding data with deep networks. In *European Conference on Computer Vision*, pp. 657–672, 2018.

A Related Work and Supplementary Analysis

A.1 Image-Level and Generative-Model Watermarking

Classical watermarking writes signals in hand-designed transform domains such as the discrete wavelet transform, discrete cosine transform, or singular value decomposition (Cox et al., 2007). Learned methods further jointly optimize an embedder, distortion layer, and reader: HiDDeN establishes an end-to-end training paradigm with differentiable noise layers (Zhu et al., 2018); StegaStamp targets geometric and physical distortions (Tancik et al., 2020); and RivaGAN improves spatiotemporal robustness through attention and adversarial training (Zhang et al., 2019). TrustMark supports arbitrary resolutions and various conventional distortions (Agarwal et al., 2025), while Watermark Anything extends watermarking to local regions and tampering scenarios (Sander et al., 2025). These methods do not depend on the source generator and are suitable for directly protecting existing images, but their primary objective remains message extraction from a single image after prescribed post-processing.

Watermarks native to generative models write signals into the generation pipeline itself. Stable Signature fine-tunes a VAE decoder to produce a fixed signature (Fernandez et al., 2023); Tree-Ring constructs a structured pattern in the Fourier domain of initial diffusion noise (Wen et al., 2023); and WOUAF uses weight modulation for user-level model attribution (Kim et al., 2023). Other work establishes verification signals through approximately Gaussian initial noise or secret-noise correlations (Hu et al., 2026; Goren et al., 2026). These methods primarily authenticate whether an image came from a controlled generator. Existing systematic evaluations show that detectability in source-model outputs does not ensure that the signal can pass through training data into a student model (Dubiński et al., 2025).

A.2 Training-Data Tracing and Radioactive Watermarking

Radioactive Data uses directionally consistent sample modifications to leave a statistically testable trace in a trained classifier (Sablayrolles et al., 2020). For generative models, invisible signals repeated across samples may likewise be inherited by the model (Ditria & Drummond, 2023). One family of methods directly modifies completed images and therefore does not require control over the process that originally generated them. DiffusionShield assigns each owner a repeated blockwise multibit pattern and jointly optimizes its basis patches and decoder, explicitly motivating downstream reproduction through pattern uniformity (Cui et al., 2023). CoprGuard embeds a watermark image in the DWT domain and combines an invertible hiding network with AutoencoderKL reconstruction and an information-enhancement module to compensate for codec degradation (Liu et al., 2025).

FT-Shield instead searches for image-dependent perturbations that are readily absorbed during fine-tuning by alternating perturbation optimization with proxy model updates; a mixture of detectors handles different fine-tuning procedures (Cui et al., 2024). SIREN optimizes a task-aligned image coating with a surrogate diffusion model, perceptual constraints, and a hypersphere detector, then uses a distributional hypothesis test on black-box generations (Li et al., 2025a). ProMark targets concept-level causal attribution (Asnani et al., 2024), while trigger-based methods bind an image coating to a text trigger or use robust templates to induce memorization (Wang et al., 2024; Ren et al., 2024). These data-side approaches can protect existing images, but use different pixel, frequency, proxy-model, or task-specific interfaces. Their propagation can vary with the fine-tuning algorithm, data domain, and removal attack, and cannot be inferred from decoding accuracy on the original protected images alone (Wang et al., 2025).

A.3 Radioactive Mechanisms Tied to Generative Representations

Another family creates radioactive images inside a controlled source generator. HMark maps a multibit secret to a shared residual in the semantic h-space of a diffusion U-Net. The residual is injected during the final reverse-diffusion step, and a pixel-space decoder predicts both presence and message bits after downstream fine-tuning (Li et al., 2025b). WIAR biases dynamically constructed green-token lists during autoregressive image generation and detects the resulting excess with a statistical test (Meintz et al., 2025). BitMark adapts this principle to bitwise autoregressive models by biasing key-specified bit patterns, whose dense statistical signal can remain detectable after cross-model training (Kerner et al., 2025). SERUM mixes a fixed normalized pattern into the initial diffusion noise and trains a lightweight latent-space detector, evaluating propagation after LoRA and full fine-tuning as well as multi-user identification (Kociszewski et al., 2026).

These generation-native approaches align their signals with the source model's representations or sampling procedure, but require access to its hidden states, token or bit logits, or initial noise. Under their reported interfaces, they do not directly watermark an already completed image. Cross-sample consistency is also a shared principle rather than a RepMark-specific premise: DiffusionShield emphasizes pattern uniformity, HMark reuses one h-space residual for a fixed secret, and SERUM reuses fixed noise patterns. RepMark combines this principle with a frozen VAE writing representation, a separate private verification space, and explicit recovery training through a different frozen VAE. The same message representation can therefore protect existing images, operate inside a compatible generator, and support post-training model attribution. Presence detection measures whether evidence propagated, while per-bit aggregation over multiple outputs tests recovery of the registered owner message.

Perceptual distortion models can constrain modifications in visually sensitive regions (Chou & Li, 1995), and quantization index modulation can establish a decision margin under bounded perturbations in specified carrier coordinates (Chen & Wornell, 2001), but neither property alone ensures that a student model will learn the signal. RepMark therefore evaluates cross-representation recovery, cross-sample consistency, and final verification on student-model outputs separately, rather than substituting any intermediate property for model-level radioactivity (Lu et al., 2026).

A.4 Conceptual Comparison with Related Watermarks

Table 5 compares the native writing interfaces, shared signal structures, training requirements, and verification scopes of related methods. It describes mechanisms and reported evaluation settings; numerical results from different papers are not treated as a matched performance comparison.

Table 5: Conceptual comparison of related watermarking methods. Student-transfer settings describe evaluations reported by the cited papers, while the RepMark row refers to this work. A method's original verification scope does not imply that other uses are impossible.

Method	Writing space and shared signal	Required interface and training	Verification and reported scope
Stable Signature (Fernandez et al., 2023)	VAE decoder parameters; one fixed bit signature per adapted decoder.	Fine-tune the source VAE decoder against a pretrained, frozen image extractor.	Image-space bit recovery and a signature test; attribution to the source generator.
HMark (Li et al., 2025b)	U-Net semantic h-space; the same secret induces a shared latent residual.	Access the U-Net bottleneck; train a secret encoder and image decoder with distortion-aware losses.	Image-space presence and bits; transfer after pixel-DDPM U-Net and LoRA fine-tuning.
SERUM (Kociszewski et al., 2026)	Initial diffusion noise; fixed normalized patterns shared across samples.	Control the initial noise; keep the generator fixed and train latent-space detectors with augmentations.	Pattern detection after VAE encoding; LoRA/full fine-tuning transfer and multi-user identification.
DiffusionShield (Cui et al., 2023)	Pixel space; reusable basis patches form an owner-specific block pattern.	Add patches to existing images; jointly optimize the patches and their decoder.	Blockwise message decoding; inheritance through training on protected images.
CoprGuard (Liu et al., 2025)	DWT frequency domain; an invertible network hides a watermark image.	Train invertible hiding/recovery and an enhancement module with AutoencoderKL reconstructions.	Recover a watermark image; similarity and multi-image tests for training/fine-tuning tracing.
RepMark (this work)	Frozen VAE latent; a bounded, carrier-independent displacement for each multibit message.	Train the writer, bilateral LoRA, and verifier through direct and frozen proxy-VAE paths.	Private-encoder presence and bits; fine-tuning/from-scratch transfer, held-out VAE, and query aggregation.

The comparison highlights two boundaries. First, cross-sample consistency is a shared principle: DiffusionShield promotes pattern uniformity, HMark uses a common h-space residual for a fixed secret, and SERUM reuses noise patterns. Second, data-side methods can modify completed images, whereas HMark, WIAR, BitMark, and SERUM rely on access to a particular generation interface. RepMark connects these settings through a frozen VAE writing representation and bilateral low-rank adaptation. Its proxy VAE is a codec channel rather than a surrogate student trained inside the watermark objective. CoprGuard also addresses codec degradation through AutoencoderKL reconstruction and an information-enhancement module; RepMark further separates shared writing from private verification and tests transfer through a held-out VAE. Its distinctive claim is therefore this combination and full-path interface, not cross-sample consistency or codec-aware recovery in isolation.

A.5 Supplementary Analysis of Shared Coordinates and Gradient Coherence

Per-image SVD gives each carrier x_i a different singular basis U_i, V_i. Even when the same message is embedded, its pixel perturbation is determined by $U_i \Delta_k(b) V_i^\top$ and may be affected by singular-vector sign flips, subspace rotations among nearly repeated singular values, and changes in mode ordering. A VAE instead uses globally shared parameters to establish common coordinates across samples, allowing the same message component $K_k b$ to recur at dataset scale.

Let the training-gradient difference produced by a matched watermarked and clean sample be $g_i^w = \nabla_\theta \ell(\theta; x_i^w) - \nabla_\theta \ell(\theta; x_i)$. Its coherence can be written as

$$\Gamma_w = \frac{\left\| \sum_{i=1}^N g_i^w \right\|_2}{\left(\sum_{i=1}^N \|g_i^w\|_2^2 \right)^{1/2}}. \tag{17}$$

When gradients are approximately zero-mean and mutually incoherent, $\Gamma_w \approx 1$; when their directions are perfectly aligned, $\Gamma_w = \sqrt{N}$. Shared coordinates therefore provide a structural condition for watermark gradients to accumulate across samples, but do not by themselves guarantee radioactivity. We observe this structure in the representation-coherence experiment and test final propagation using novel samples generated by downstream models.

B Experimental Details

B.1 RepMark Training

We train RepMark on 5,000 images generated by FLUX.2-dev using prompts sampled from Stable-Diffusion-Prompts. Training resolutions are 256^2, 512^2, and 1024^2. Images retain their original aspect ratios: the long side is resized to the target resolution, and dimensions are padded only to multiples of 16. A separate 16-bit message is sampled for every example during watermark training; data-protection and radioactivity experiments instead use one fixed owner message. The base FLUX.2-dev VAE remains frozen. We update only the encoder and decoder LoRA modules, watermark writer, feature network, weighted aggregation network, message-extraction head, and presence-detection head. Training uses Adam with cosine learning-rate decay. Learning rates are 10^{-5} for LoRA and 5×10^{-5} for the remaining watermark modules. We do not apply JPEG, cropping, blur, random noise, or other conventional image perturbations during training. The cross-representation branch performs a frozen SDXL VAE round trip and is optimized jointly with the direct path.

Table 6: Main RepMark training configuration.

Configuration	Value
Training images	5,000
Training resolutions	256 / 512 / 1024
Message length L	16
FLUX.2-dev latent channels	128
LoRA rank / scale	16 / 8
LoRA learning rate	10^{-5}
Writer and verifier learning rate	5×10^{-5}
Proxy representation path	Frozen SDXL VAE
Presence top-K fraction	0.1

Image-quality loss. The grouped terms in Equation (14) comprise the following concrete weighted sub-losses. Let $r = x^w - x$ denote the residual between the raw watermarked output and its original image. The quality group is

$$\begin{aligned} \mathcal{L}_{\text{qual}} = \lambda_{\text{mse}} \, \text{MSE}(x^w, x) + \lambda_{\text{jnd}} \mathcal{L}_{\text{jnd}} + \lambda_{\text{fft}} \mathcal{L}_{\text{fft}} \\ + \lambda_{\text{mean}} \mathcal{L}_{\text{mean}} + \lambda_{\text{lpips}} \mathcal{L}_{\text{lpips}} + \lambda_{\text{latent}} \, \text{MSE}(z^w, z). \end{aligned} \tag{18}$$

Here, MSE is averaged over all tensor elements and $z^w - z = \delta_b$. Let $M(x)$ be the JND mask computed from the original image. The normalized JND-weighted reconstruction term is

$$\mathcal{L}_{\text{jnd}} = \text{mean}\left(\frac{1 + \gamma M(x)}{\text{mean}(1 + \gamma M(x))} \odot r^2 \right), \qquad \gamma = 1. \tag{19}$$

$\mathcal{L}_{\text{fft}}$ penalizes local anomalous peaks in the residual spectrum. Specifically, let $P = |\,\text{FFT}_{\text{ortho}}(r)|^2$, and let $\overline{P}_p$ be the mean power in the cyclic 5×5 neighborhood around frequency bin p, excluding the center. Then

$$\mathcal{L}_{\text{fft}} = \text{mean}_{p \neq (0,0)} \left\{ \min\left(\frac{P_p}{\text{mean}(P)}, 10 \right) \left[\log(P_p + \varepsilon) - \log(\overline{P}_p + \varepsilon) - \log 4 \right]_+^2 \right\}. \tag{20}$$

The DC bin is excluded from this average. $\mathcal{L}_{\text{mean}} = \text{mean}_{n,c} \Big[(HW)^{-1} \sum_{h,w} r_{nchw} \Big]^2$ penalizes the mean residual bias of each channel in each image, $\mathcal{L}_{\text{lpips}}$ constrains perceptual feature distance, and the final term regularizes the latent writing displacement. All image-quality terms above are computed before the optional proxy-VAE verification path.

Message-recovery loss. For extraction logit e_i and target bit b_i, let $\ell_i = \text{BCEWithLogits}(e_i, b_i)$. We combine the mean loss over all bits with a hard-bit term that averages, for each image, the K largest bit losses:

$$\mathcal{L}_{\text{msg}} = (1 - \rho) \frac{1}{L} \sum_{i=1}^{L} \ell_i + \rho \frac{1}{K} \sum_{i \in \text{TopK}(\ell)} \ell_i. \tag{21}$$

We use $\rho = 0.3$ and $K = 4$. Because matched pairs are stochastically assigned to either the direct route or the proxy-VAE route, this objective supervises message extraction under both routes over training.

Presence-detection loss. Each watermarked sample forms a pair with its matched clean negative. For a logit a and label $y \in \{0, 1\}$, define the positive-class-weighted binary cross-entropy as $\text{BCE}_w(a, y) = -wy \log \sigma(a) - (1 - y) \log(1 - \sigma(a))$. For a batch of B pairs, let d_n^+, d_n^- be the global logits of pair n and let $u_{n,i}^+, u_{n,i}^-$ be its presence logits for bit i. Then

$$\mathcal{L}_{\text{global-BCE}} = \frac{1}{2B} \sum_{n=1}^{B} \left[\text{BCE}_w(d_n^+, 1) + \text{BCE}_w(d_n^-, 0) \right], \tag{22}$$

$$\mathcal{L}_{\text{bit-BCE}} = \frac{1}{2BL} \sum_{n=1}^{B} \sum_{i=1}^{L} \left[\text{BCE}_w(u_{n,i}^+, 1) + \text{BCE}_w(u_{n,i}^-, 0) \right]. \tag{23}$$

The presence-detection loss combines the global and per-bit BCE terms with their corresponding paired margins:

$$\begin{aligned} \mathcal{L}_{\text{det}} = \lambda_{\text{detect}} \mathcal{L}_{\text{global-BCE}} + \lambda_{\text{margin}} \frac{1}{B} \sum_{n=1}^{B} \left[m_g - (d_n^+ - d_n^-) \right]_+ \\ + \lambda_{\text{bit-presence}} \mathcal{L}_{\text{bit-BCE}} + \lambda_{\text{bit-margin}} \frac{1}{BL} \sum_{n=1}^{B} \sum_{i=1}^{L} \left[m_b - (u_{n,i}^+ - u_{n,i}^-) \right]_+ . \end{aligned} \tag{24}$$

Here, $[a]_+ = \max(a, 0)$. Every per-bit presence logit inherits the image-level presence label; this supervision indicates whether structured evidence exists in that channel and is distinct from the zero/one message-bit target used by Equation (21). The positive-class weight is 2, and the global and per-bit margins are $m_g = 2$ and $m_b = 1$, respectively.

Paired sample construction and weights. The positive member of each pair is x^w. Its negative is sampled with equal probability from the original image x and the no-displacement reconstruction $\mathcal{D}_\phi(\mathcal{E}_0(x))$. With probability 0.2, both members of a pair undergo the same frozen SDXL-VAE encode–decode round trip; otherwise both follow the direct route. Pairwise routing prevents the detector from using the presence of the proxy-codec transformation as a shortcut.

The sub-loss weights are $\lambda_{\text{mse}} = 10$, $\lambda_{\text{jnd}} = 2$, $\lambda_{\text{fft}} = 0.5$, $\lambda_{\text{mean}} = 2$, $\lambda_{\text{lpips}} = 1$, $\lambda_{\text{extract}} = 0.5$, $\lambda_{\text{detect}} = 0.01$, $\lambda_{\text{margin}} = 0.25$, $\lambda_{\text{bit-presence}} = 0.5$, $\lambda_{\text{bit-margin}} = 0.1$, and $\lambda_{\text{latent}} = 0.01$. Here λ_{extract} multiplies Equation (21); the remaining named coefficients multiply the corresponding terms in Equations (18) and (24).

B.2 Baseline Configuration

We compare DWT–DCT and DWT–DCT–SVD (Cox et al., 2007; ShieldMnt, 2020), RivaGAN (Zhang et al., 2019), TrustMark-Q (Agarwal et al., 2025), Watermark Anything (Sander et al., 2025), DiffusionShield (Cui et al., 2023), and CoprGuard (Liu et al., 2025). The first two use the `invisible-watermark` implementation; RivaGAN uses its public pretrained model; TrustMark-Q and Watermark Anything use their official models; and DiffusionShield and CoprGuard are integrated from their public implementations under the same data pipeline. Every method retains its native writer, message length, and presence score, and no detector is retrained. RivaGAN, the DWT methods, and Watermark Anything use 32-bit messages; TrustMark-Q uses its native BCH-coded message; DiffusionShield uses a 128-bit message; and CoprGuard provides only a keyed presence score and therefore has no BitAcc result. HMark and SERUM provide neither an implementation nor model weights that can be applied under our protocol, so they are discussed in related work but excluded from quantitative comparison.

B.3 Evaluation Metrics

Image-level verification. Detection rate is the fraction of samples above a method's fixed operating threshold, and image-level BitAcc is the mean per-bit accuracy over all watermarked images. Primary requires a positive presence decision and $\text{BitAcc} \geq 0.75$, while Exact recovery requires a positive presence decision and every message bit to be correct. Because methods use different message lengths and native decision rules, cross-method comparison primarily relies on presence-score AUROC and TPR@1%FPR. The TPR threshold is chosen using a disjoint clean calibration set and remains fixed under every attack condition. CoprGuard participates only in comparisons of presence metrics.

Image-quality and representation metrics. We report LPIPS, SSIM, CLIPScore, and FID in the main text, with PSNR and KID in the supplement. Directional coherence C measures the consistency of paired displacements, while local mixing M_{200} measures neighborhood mixing between clean and watermarked examples; M_{200} near 0.5 indicates balanced mixing. Linear AUROC uses image-identity-grouped five-fold cross-validation, and t-SNE is used only for visualization. The experiments train separate student-model instances for different methods, resolutions, training budgets, protected-data fractions, and representation settings. Each protected student is compared with a separately trained clean-reference student using the same architecture, data, and training protocol, with watermarking as the controlled difference. Model-level metrics are first computed within each of three independent student-training seeds and then averaged across seeds; outputs from different students are never pooled. Per-image AUROC and TPR measure presence in each student's output distribution, while the supplementary query-set experiment averages presence scores over Q outputs of that student and applies a per-bit hard majority vote to obtain query-aggregated BitAcc.

B.4 Model-Level Radioactivity Protocols

PixArt fine-tuning. We write a fixed owner message into training images and fine-tune a separate PixArt-Σ student for each method–resolution setting. The main RepMark setting uses 500 protected training images, global batch size 4, learning rate 10^{-5}, and 1,000 optimization steps at each of resolutions 256, 512, and 1024. For each of three independent student-training seeds, the protected and clean-reference students are evaluated on 100 held-out prompts with five generation seeds per prompt, producing 500 outputs per student. The generation seeds provide repeated outputs within a trained student and are distinct from the three student-training seeds. Generation uses 20 sampling steps and CFG 4.5. The cross-method retest evaluates each separately trained student on 500 images. AUROC confidence intervals use 2,000 prompt-clustered bootstrap replicates.

The protected-fraction experiment trains a separate student for every configuration, keeps the total training set fixed at 500 images, and tests 0%, 10%, 25%, 50%, 75%, and 100% protection at reso-

lution 1024. The training-set-size experiment likewise trains separate students at a 100% protected fraction using 125, 250, 500, and 1,000 training images. Both use 1,000 optimization steps. Separately, on a fixed 200K-image training set, we train a randomly initialized LightningDiT model for every method–fraction configuration, covering six methods and protected fractions of 5%, 10%, 20%, 30%, 50%, 70%, and 90%, to 40K steps.

Training from scratch. The unified comparison uses 2M images from the KempnerInstituteAI dataset at 256^2 and native FLUX.2-dev VAE latents to train a separate LightningDiT-XL/1 student from random initialization for every method–training-budget setting and student-training seed, together with its clean-reference counterpart. All students share global batch size 128 and learning rate 5×10^{-4}; the evaluated budgets are 10K, 20K, 40K, 80K, and 100K steps. Each of the three independent training seeds is evaluated on 512 generations using generation seed 4242, 250 sampling steps, and CFG 1.0. Reported model-level values average the three seed-specific metrics. The external-representation experiment uses the SD3.5-large VAE, which is excluded from watermark training, to encode 500K protected training images into 16-channel latents and separately trains LightningDiT-XL/2 students from scratch at resolutions 256 and 512 for 80K steps.

B.5 Representation and Concealment Protocols

Latent representations. For 1,000 content-matched clean/watermarked pairs, we extract representations from the original FLUX.2-dev VAE, LoRA-adapted FLUX.2-dev VAE, the proxy SDXL VAE used during training, and the external SD3.5 VAE. Each latent descriptor concatenates a 4×4 adaptive average pool with per-channel mean, standard deviation, RMS, maximum, and minimum. The four descriptor dimensions are 2,688, 2,688, 84, and 336. Directional coherence is computed on paired displacements after per-dimension variance standardization. Local mixing jointly standardizes the features, reduces them to 50 dimensions with PCA, applies per-sample L_2 normalization, excludes each sample and its matched counterpart, and retrieves 200 nearest neighbors.

Pixel and frequency representations. Pixel and frequency experiments use 5,000 pairs at 256^2. The pixel descriptor concatenates 32×32 pooled RGB values with per-channel statistics. The frequency descriptor applies an orthonormal two-dimensional FFT to every RGB channel, computes $\log(1 + |\,\mathrm{FFT}\,|)$, centers the spectrum, and concatenates pooled values and channel statistics; phase is discarded. Other methods pair each watermarked image with its original clean PNG. For RepMark, each image is encoded once and the same latent z is decoded as $\mathcal{D}_\phi(z)$ and $\mathcal{D}_\phi(z + \delta_b)$, isolating the effects of VAE reconstruction and the LoRA decoder from the watermark itself. An unwatermarked VAE round trip is reported only as a reconstruction-variability reference and is not numerically subtracted from RepMark's metrics.

Conventional post-processing. Robustness evaluation uses 128 matched watermarked/clean pairs at 1024^2 and a disjoint set of 500 clean images to calibrate the 1% FPR threshold for each method. Attacks include three JPEG levels, two blur levels, two additive-noise levels, resizing, color perturbation, and center cropping. The same threshold is used at every attack severity.

B.6 Three-Path Protocol

The three paths use matched prompts, random seeds, and base denoised latents and evaluate both the original FLUX.2-dev model and a FLUX.2-dev model fine-tuned on protected training data. The native path directly decodes the denoised latent without an additional message. The post-processing path first saves the native RGB image, then re-encodes it with the original FLUX.2-dev encoder, adds the fixed 16-bit message, and decodes it with the LoRA decoder. The in-processing path adds the same message directly to the final denoised latent through VAE replacement. The pre-processing path is represented by native outputs of the model fine-tuned on protected training data; the post-processing and in-processing stacks test repeated writing on an already inherited signal.

Every 256^2 group contains 5,000 images and every 512^2 group contains 500. Generation uses 30 sampling steps, CFG 4.0, and fixed per-example seeds. In addition to detection rate, BitAcc, and Exact recovery, we evaluate image quality on the 256^2 paired set and measure incremental post-processing and in-processing latency after CUDA synchronization.

C Additional Experiments

This appendix reports completed results omitted from the main text, including results across training budgets, representation and concealment analyses, deployment diagnostics, image-level robustness, semantic editing, and design-evolution ablations. We do not repeat endpoint values already presented in the main paper.

C.1 Complete Model-Level Radioactivity Results

Complete 500-image PixArt retest. Table 7 supplements the three AUROC values retained in the main comparison with TPR, native detection rate, BitAcc, and clean FPR. Each method–resolution cell reports the mean over three independently trained protected/clean PixArt student pairs. Per student, evaluation uses 500 generations from held-out prompts (100 prompts $\times$ 5 generation seeds) and 500 clean-student outputs generated with the same prompts and generation seeds. AUROC intervals use 2,000 prompt-clustered bootstrap replicates; for 0/500 false positives, we report the Wilson upper bound.

Table 7: Complete 500-image PixArt fine-tuning retest, reported as means over three independent student-training seeds. Brackets give 95% confidence intervals. CoprGuard provides only a 1-bit presence decision and therefore has no multibit accuracy.

Resolution	Method	AUROC [95% CI]	TPR@1%FPR	Detection rate	BitAcc	Clean FPR
256	RepMark	0.999 [.999,1.000]	0.990	0.992	0.751	0.0160
	RivaGAN	0.580 [.545,.615]	0.012	0.004	0.513	$\leq$.0076
	WAM	0.536 [.499,.571]	0.042	0.000	0.500	$\leq$.0076
	DiffusionShield	0.532 [.497,.569]	0.016	0.000	0.570	$\leq$.0076
	CoprGuard	0.518 [.484,.553]	0.006	0.000	N/A	$\leq$.0076
	DWT–DCT–SVD	0.509 [.473,.545]	0.008	0.006	0.502	0.0060
	DWT–DCT	0.508 [.473,.543]	0.006	0.000	0.500	$\leq$.0076
	TrustMark-Q	0.504 [.469,.539]	0.000	0.028	0.525	$\leq$.0076
512	RepMark	1.000 [1,1]	1.000	1.000	0.763	$\leq$.0076
	CoprGuard	0.721 [.691,.750]	0.036	0.000	N/A	$\leq$.0076
	RivaGAN	0.719 [.688,.748]	0.020	0.006	0.539	$\leq$.0076
	WAM	0.715 [.684,.746]	0.142	0.000	0.524	$\leq$.0076
	DiffusionShield	0.567 [.533,.603]	0.008	0.000	0.570	$\leq$.0076
	DWT–DCT–SVD	0.529 [.495,.564]	0.048	0.018	0.506	$\leq$.0076
	TrustMark-Q	0.471 [.437,.507]	0.008	0.030	0.526	$\leq$.0076
1024	RepMark	1.000 [1,1]	1.000	1.000	0.830	$\leq$.0076
	WAM	0.777 [.748,.805]	0.148	0.000	0.500	$\leq$.0076
	RivaGAN	0.729 [.698,.759]	0.076	0.010	0.565	$\leq$.0076
	CoprGuard	0.598 [.564,.632]	0.014	0.000	N/A	$\leq$.0076
	DWT–DCT–SVD	0.533 [.500,.567]	0.084	0.034	0.509	$\leq$.0076
	DWT–DCT	0.512 [.482,.543]	0.008	0.000	0.501	$\leq$.0076
	TrustMark-Q	0.511 [.474,.548]	0.004	0.020	0.529	$\leq$.0076

Complete from-scratch results across training budgets. Figure 6 reports five training budgets for six methods. Each method–budget setting reports the mean over three independently trained LightningDiT-XL/1 students and their clean-reference counterparts, all using the same 2,282,665 training images at 256 pixels, native FLUX.2-dev VAE latents, and random-initialization protocol. Every student is evaluated independently on 512 generations.

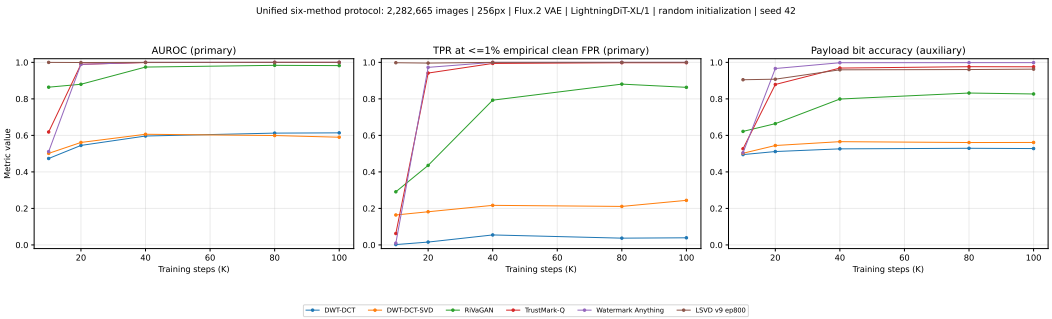


Figure 6: Complete from-scratch results across five training budgets for the six methods in the main comparison. From left to right: AUROC, TPR at no more than 1% empirical clean FPR, and auxiliary BitAcc. Curves report means over three independent student-training seeds under the same protocol.

RepMark maintains near-perfect AUROC and TPR from 10K steps. TrustMark-Q and Watermark Anything establish inheritance rapidly after 20K, RivaGAN improves gradually, and both DWT variants remain only weakly distinguishable throughout.

Protected fraction over training. Figures 4 and 5 in the main text report dose responses at fixed training budgets. Figure 7 gives the complete results across training budgets for all six methods and seven protected-data fractions. Each method–fraction–budget setting uses a separately trained student.

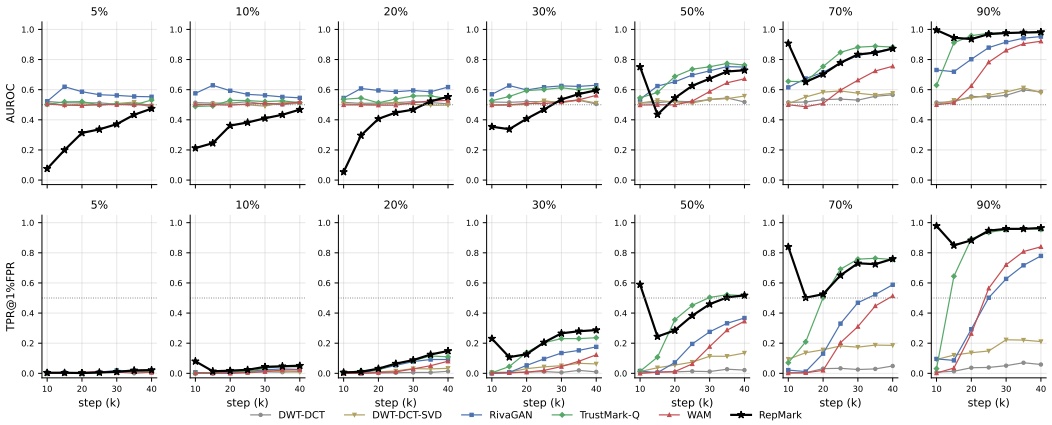


Figure 7: AUROC (top) and TPR@1%FPR (bottom) over training at each protected fraction for LightningDiT trained from scratch at 256 pixels. RepMark rises early at medium and high fractions; classical frequency-domain methods remain close to chance.

Across-model evaluation and within-model query aggregation. The main experiments evaluate separately trained students across methods, resolutions, protected-data fractions, and representation settings. Within each trained student, we isolate the effect of query count by reusing its per-image records and forming each query set from Q of its outputs. The presence score is the mean of the Q per-image presence scores. For message verification, we apply a per-bit hard majority vote to the Q decoded messages and compute BitAcc according to Equation (15). From outputs of students trained on protected training data and clean reference images, we draw 20,000 query sets without replacement and set the 1% FPR threshold to the 99th percentile of the clean query-set scores. Table 8 and Figure 8 report the results.

Table 8: Multi-image query aggregation performed independently within each trained student instance. PixArt reports query-set AUROC, TPR@1%FPR, and mean BitAcc, where BitAcc uses a per-bit hard vote within each query set. LightningDiT reports query-set AUROC/TPR at 40K steps for separately trained protected-fraction settings. $Q = 1$ recovers the original per-image metrics.

		PixArt fine-tuning				LightningDiT from scratch, 40K steps			
Resolution	Q	AUROC	TPR	BitAcc	Fraction	Query-set AUROC / TPR@1%FPR			
						$Q = 1$	$Q = 4$	$Q = 16$	$Q = 64$
256	1	.999	.990	.751	5%	.474/.023	.495/.079	.543/.273	.804/.739
256	4	1.000	1.000	.714	10%	.468/.051	.502/.174	.657/.520	.967/.952
256	16	1.000	1.000	.837	20%	.553/.148	.705/.471	.952/.924	1.000/1.000
256	64	1.000	1.000	.889	30%	.596/.287	.846/.742	.997/.995	1.000/1.000
512	1	1.000	1.000	.763	50%	.729/.518	.967/.945	1.000/1.000	1.000/1.000
512	4	1.000	1.000	.729	70%	.873/.760	.998/.997	1.000/1.000	1.000/1.000
512	16	1.000	1.000	.802	90%	.983/.965	1.000/1.000	1.000/1.000	1.000/1.000
512	64	1.000	1.000	.803					
1024	1	1.000	1.000	.830					
1024	4	1.000	1.000	.813					
1024	16	1.000	1.000	.889					
1024	64	1.000	1.000	.915					

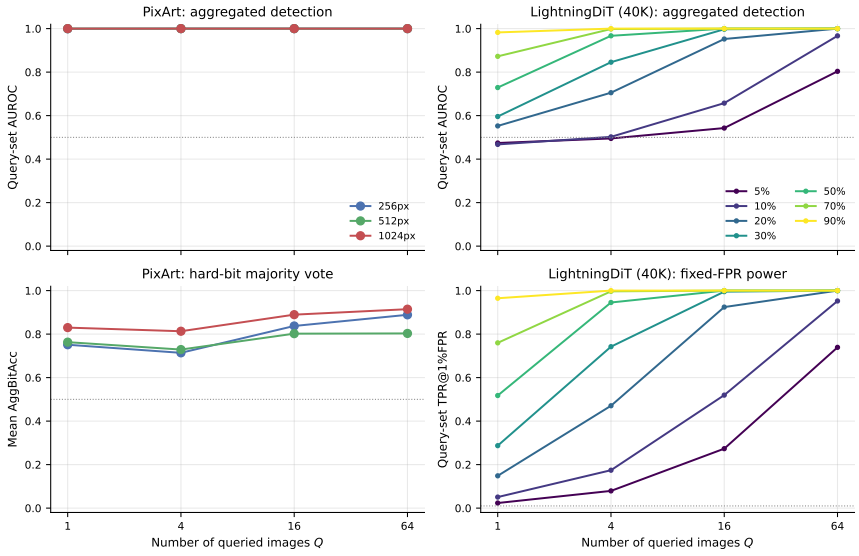


Figure 8: Effect of query count, evaluated independently within each trained student. Left: query-set AUROC and hard-majority-vote BitAcc for separately trained PixArt students at three resolutions. Right: query-set AUROC and TPR@1%FPR for separately trained LightningDiT protected-fraction settings at 40K steps.

PixArt already reaches 0.999–1.000 per-image presence AUROC, so query-set presence detection saturates by $Q = 4$; aggregation mainly benefits message extraction. At 1024^2, mean BitAcc rises from 0.830 at $Q = 1$ to 0.915 at $Q = 64$, and reaches 0.938 when all 500 outputs are voted. The 512^2 curve is not strictly monotonic, showing that hard voting suppresses random per-image errors but cannot correct bit positions that are systematically wrong across images. The weak-fraction LightningDiT setting more clearly shows the aggregation gain: at a 20% protected fraction, query-set AUROC rises from 0.553 for $Q = 1$ to 0.952 for $Q = 16$ and 1.000 for $Q = 64$; TPR rises from 0.148 to 0.924 and 1.000.

The 20,000 query sets in each condition are repeatedly sampled from that student's finite output pool and its clean-reference pool, so query sets may overlap. This analysis quantifies query-sampling variation separately for every trained student; it complements, rather than replaces or inflates, the across-model evaluation over independently trained student instances.

C.2 Representation and Concealment Analyses

C.2.1 Latent Statistics

The main text reports directional coherence C and local mixing M_{200}. Table 9 adds relative displacement Δ_{rel} and image-grouped linear AUROC for the same 1,000 images in three latent spaces. Every statistic is computed in the original high-dimensional feature space.

Table 9: Relative latent displacement and linear AUROC omitted from the main representation table. SDXL is the proxy VAE used during training.

	Original FLUX.2-dev		LoRA-adapted FLUX.2-dev		Proxy SDXL VAE	
Method	Δ_{rel}	AUROC	Δ_{rel}	AUROC	Δ_{rel}	AUROC
RivaGAN	0.163	0.899	0.174	0.867	0.057	0.520
DWT–DCT	0.135	0.787	0.218	0.852	0.064	0.540
DWT–DCT–SVD	0.139	0.769	0.176	0.754	0.065	0.552
TrustMark-Q	0.118	0.658	0.216	0.714	0.087	0.537
Watermark Anything	0.212	0.977	0.290	0.993	0.071	0.564
RepMark	**0.251**	**1.000**	**1.195**	**1.000**	**0.126**	**0.641**

C.2.2 Image Quality and Public Representations

Additional quality metrics. Table 10 supplements LPIPS, SSIM, CLIPScore, and FID with PSNR, KID, and three paired preference changes. RepMark uses an unwatermarked same-latent reconstruction from the same LoRA decoder as reference, eliminating VAE posterior sampling and base reconstruction error so that only the net change caused by the watermark displacement remains; other post-processing methods use the matched original clean PNG as reference.

Table 10: Additional image-quality metrics at 256 pixels on 5,000 pairs. KID is reported as $\times 1000$; ΔCLIP, ΔIR, and ΔHPS are paired changes in CLIPScore, ImageReward, and HPSv2.

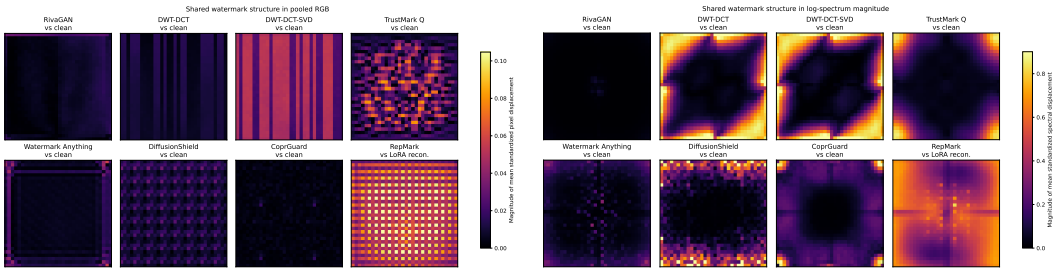
Method	PSNR↑	KID↓	ΔCLIP↑	ΔIR↑	ΔHPS↑
RivaGAN	42.188	0.8495	+0.1609	+0.0157	-0.003542
DWT–DCT	34.234	0.2745	-0.1609	+0.0197	-0.002375
DWT–DCT–SVD	34.186	0.3803	-0.2883	+0.0138	-0.003441
TrustMark-Q	40.328	0.1467	+0.0233	-0.0012	-0.001540
Watermark Anything	35.993	1.5951	-0.0577	-0.0003	-0.006222
DiffusionShield	31.826	1.5118	-0.4629	-0.0072	-0.001590
CoprGuard	37.335	-0.1855	-0.2317	-0.0031	-0.000926
RepMark	35.501	0.1519	-0.1943	+0.0021	+0.003618

Public separability. The main text reports C and M_{200}. Table 11 adds relative displacement and image-grouped five-fold linear AUROC in public pixel and spectrum features. All statistics use the latest 5,000-image experiment; RepMark compares $D_{\mathrm{LoRA}}(z)$ with $D_{\mathrm{LoRA}}(z+\delta)$.

Table 11: Relative displacement and linear AUROC in public RGB and log-spectrum-magnitude representations on 5,000 pairs.

	Pixel RGB		Spectrum magnitude	
Method	Δ_{rel}	AUROC	Δ_{rel}	AUROC
RivaGAN	0.016	0.517	0.027	0.526
DWT–DCT	0.010	0.509	0.245	0.983
DWT–DCT–SVD	0.027	0.539	0.240	0.986
TrustMark-Q	0.022	0.509	0.127	0.868
Watermark Anything	0.021	0.512	0.064	0.691
DiffusionShield	0.010	0.513	0.140	0.998
CoprGuard	0.008	0.503	0.157	0.971
RepMark	0.043	0.540	0.255	0.697

Figure 9 shows the spatial and frequency-band structure of the same statistics. Methods share a color scale within each panel group, and RepMark uses the same-latent LoRA reconstruction as its reference.



(a) Mean standardized displacement in pooled RGB.

(b) Mean standardized displacement in log-spectrum magnitude.

Figure 9: Cross-sample common displacement of eight methods in public pixel and frequency representations.

Frequency-band distribution. Figure 10 gives radial spectrum-displacement profiles for eight methods, showing how their common response varies with spatial frequency.

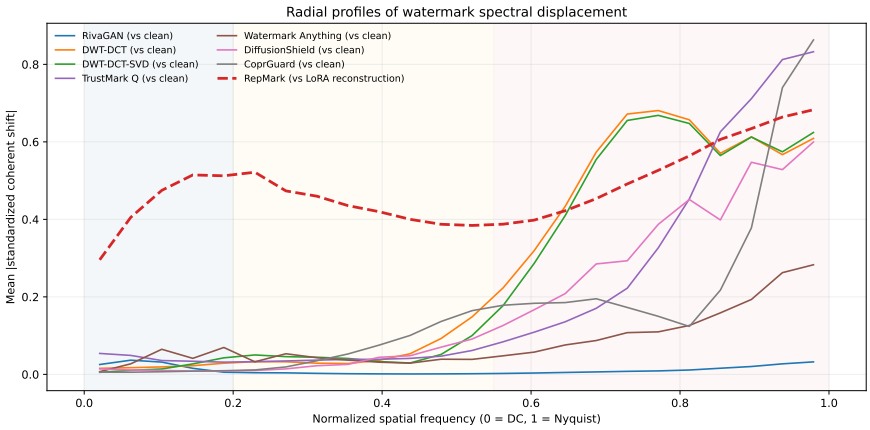


Figure 10: Radial mean standardized displacement in log-spectrum magnitude for eight watermarking methods.

Same-latent reconstruction control. A deterministic round trip through the original FLUX.2-dev VAE yields LPIPS 0.0052, SSIM 0.9804, PSNR 35.985, FID 0.5069, and KID 0.0882. Across 30 posterior samples, dataset means range from $[0.010840, 0.010877]$ for LPIPS, $[0.7787, 0.7959]$ for FID, $[-0.3029, 0.1243]$ for KID, and $[0.1049, 0.1275]$ for ΔCLIP. These values characterize stochastic VAE reconstruction variability and are not subtracted from the same-latent RepMark measurement.

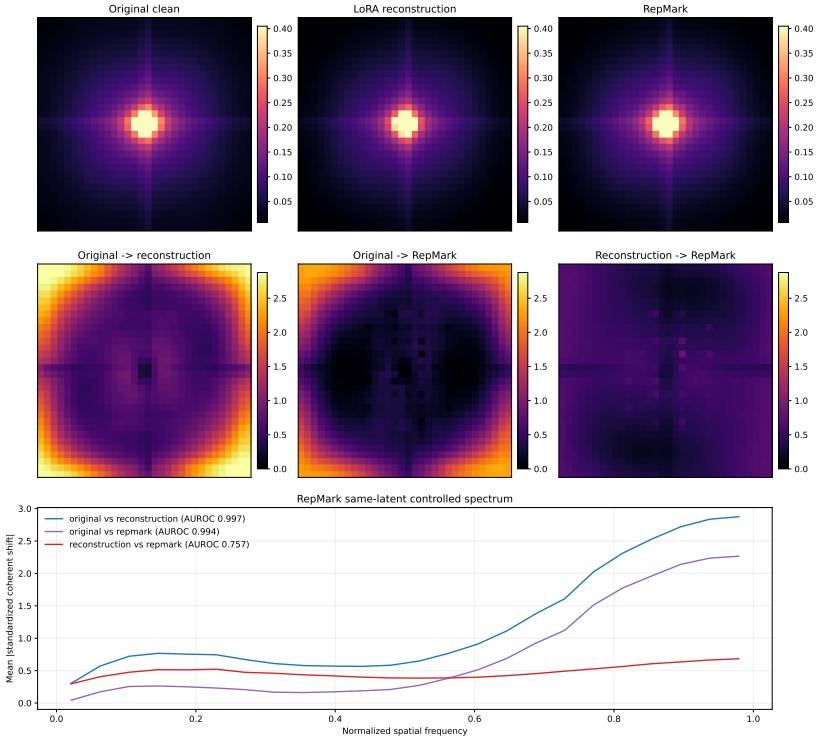


Figure 11: Same-latent control for RepMark. Top: mean spectra of original images, unwatermarked LoRA reconstructions, and RepMark images. Middle: reconstruction and net watermark displacement. Bottom: the corresponding radial profiles.

Visual comparisons across methods. Figures 12 and 13 show four original images after watermarking and their absolute pixel residuals. Each example includes the watermarked image, the original residual, and the residual magnified by $4\times$. The Clean column compares an unwatermarked FLUX.2-dev VAE reconstruction with the original; other columns compare each protected image with the same original. All methods use their native operating points. Residuals share a fixed display gain, without per-image normalization.



Figure 12: Paired visual comparisons at 256^2: a person and a forest. The first column is clean; the remaining columns are RivaGAN, DWT–DCT, DWT–DCT–SVD, TrustMark-Q, Watermark Anything, DiffusionShield, CoprGuard, and RepMark. Under each example are absolute residuals at their original scale and with a shared $4\times$ gain. Clean-column residuals show unwatermarked VAE reconstruction error.

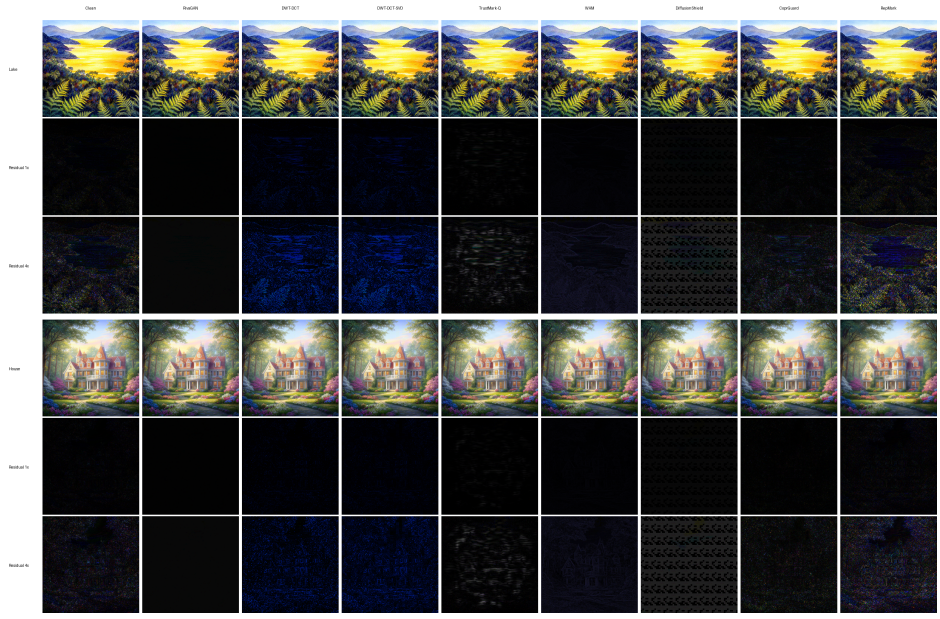


Figure 13: Paired visual comparisons at 256^2: a lake and a residential landscape. Column order and residual displays follow Figure 12.

C.3 Deployment Diagnostics

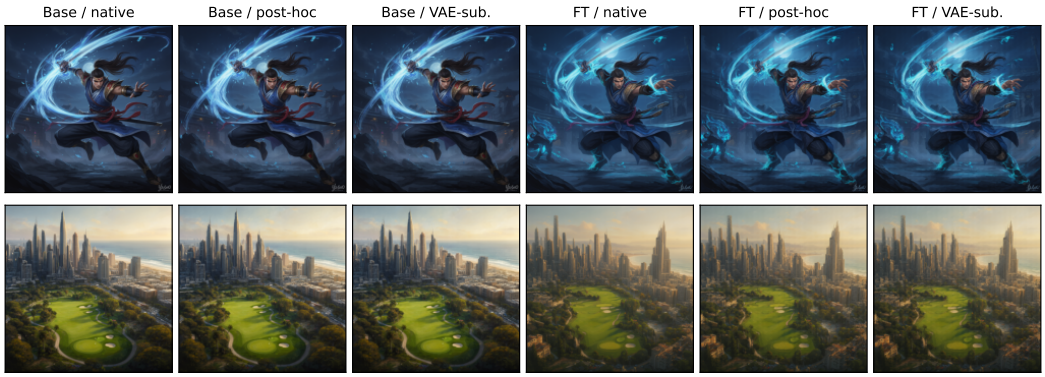


Figure 14: Qualitative results for the three deployment paths on the original model and the model fine-tuned on protected training data at 512^2. Columns in each row share the same prompt, random seed, and base denoised latent. From left to right: native, post-processing, and in-processing outputs.

In-processing through VAE replacement writes directly to the final denoised latent and avoids re-encoding a completed image. In the 5,000-image evaluation, incremental latency for post-processing/in-processing is 24.58/17.35 ms on the original model and 24.67/17.43 ms on the model fine-tuned on protected data. In-processing is approximately 29% faster.

Native outputs of the fine-tuned model already inherit the fixed message. Writing it again systematically interferes with particular bits: bits 9 and 13 flip from correct to incorrect 2,249/4,816 times under post-processing and 2,361/4,857 times under in-processing. Both paths still retain a 100% detection rate, showing that repeated writing primarily affects exact recovery rather than presence.

C.4 Image-Level Robustness

Conventional post-processing. We evaluate eight methods on the same 1024^2 images. The standard matrix covers three JPEG levels, two blur levels, two noise levels, resizing, color perturbation, and unresized center cropping. Each method–attack cell contains 128 matched watermarked/clean pairs. CoprGuard provides only a 1-bit presence score and therefore has no BitAcc.

Table 12: Payload recovery under conventional post-processing (BitAcc, %). CropNR.7 is a center crop retaining 0.7 of the image without resizing back.

Method	None	8bit	J50	J20	J10	CropNR.7	Resize	Blur3	Blur6	N.08	N.25	Color
RepMark	100.0	100.0	93.7	76.9	67.7	100.0	99.9	58.5	37.8	79.4	68.5	99.9
RivaGAN	99.8	99.8	99.6	98.1	92.0	96.4	99.8	98.5	82.6	99.5	86.9	98.6
DWT–DCT	86.1	86.1	58.7	52.2	52.5	87.0	72.4	50.6	49.7	79.1	39.8	58.4
DWT–DCT–SVD	100.0	100.0	100.0	50.1	50.0	87.3	100.0	64.8	50.2	100.0	66.9	61.4
TrustMark-Q	100.0	100.0	100.0	99.9	98.0	99.9	100.0	100.0	100.0	100.0	99.9	99.0
Watermark Anything	100.0	100.0	100.0	100.0	99.6	100.0	100.0	100.0	100.0	100.0	100.0	100.0
DiffusionShield	100.0	100.0	57.0	57.0	57.0	57.0	57.0	57.0	57.0	100.0	73.4	98.0
CoprGuard	–	–	–	–	–	–	–	–	–	–	–	–

Table 13: Watermark detection under each method's native fixed decision rule (TPR, %). CropNR.7 is a center crop retaining 0.7 of the image without resizing back.

Method	None	8bit	J50	J20	J10	CropNR.7	Resize	Blur3	Blur6	N.08	N.25	Color
RepMark	100.0	100.0	100.0	99.2	95.3	100.0	100.0	99.2	0.0	98.4	15.6	100.0
RivaGAN	100.0	100.0	100.0	100.0	97.7	100.0	100.0	100.0	96.1	100.0	100.0	100.0
DWT–DCT	78.9	78.9	30.5	10.9	12.5	99.2	55.5	2.3	0.0	66.4	0.0	30.5
DWT–DCT–SVD	100.0	100.0	100.0	0.8	0.0	100.0	100.0	32.8	0.0	100.0	68.8	39.8
TrustMark-Q	100.0	100.0	100.0	99.2	73.4	100.0	100.0	100.0	100.0	100.0	99.2	89.8
Watermark Anything	100.0	100.0	100.0	100.0	86.7	100.0	100.0	100.0	100.0	100.0	100.0	100.0
DiffusionShield	100.0	100.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	99.2	0.0	88.3
CoprGuard	100.0	100.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	100.0

Table 14: Threshold-independent presence AUROC under conventional post-processing (%). CropNR.7 is a center crop retaining 0.7 of the image without resizing back.

Method	None	8bit	J50	J20	J10	CropNR.7	Resize	Blur3	Blur6	N.08	N.25	Color
RepMark	100.0	100.0	100.0	99.7	98.6	100.0	100.0	100.0	57.1	100.0	100.0	100.0
RivaGAN	100.0	100.0	100.0	100.0	98.7	100.0	100.0	100.0	98.9	100.0	100.0	100.0
DWT–DCT	94.3	94.3	68.1	58.4	58.0	99.6	82.6	54.3	48.1	88.1	19.3	63.2
DWT–DCT–SVD	100.0	100.0	100.0	50.0	50.0	100.0	100.0	71.7	47.1	100.0	87.3	66.3
TrustMark-Q	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
Watermark Anything	100.0	100.0	100.0	100.0	99.9	100.0	100.0	100.0	100.0	100.0	100.0	100.0
DiffusionShield	100.0	100.0	56.0	52.9	52.8	100.0	100.0	52.8	51.9	100.0	100.0	100.0
CoprGuard	100.0	100.0	59.3	52.8	51.1	100.0	95.0	66.6	56.8	100.0	99.6	100.0

Because fixed-rule TPR is sensitive to attack-induced score drift, AUROC is the primary cross-method statistic. Figure 15 scans six attack families over densely sampled severities. Every horizontal axis increases in attack strength from left to right, and the dashed line marks chance AUROC of 0.5.

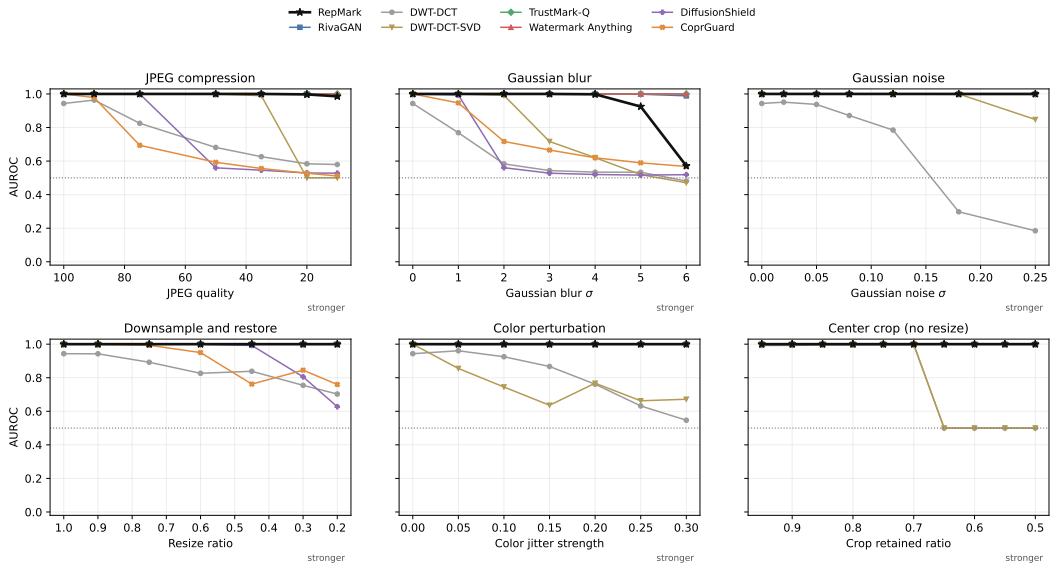


Figure 15: AUROC versus attack strength at 1024^2 for eight methods, with 128 pairs per point. JPEG quality, crop retention, and resize ratio use reversed axes; center cropping is not resized back.

RepMark retains high AUROC under JPEG, noise, resizing, color perturbation, and unresized center cropping, with its main degradation under strong blur ($\sigma = 6$). Watermark Anything and RivaGAN are stronger under conventional post-processing, illustrating that this property is distinct from model-level radioactivity.

Generative regeneration. We test SDXL and SD3.5 VAE round trips and three diffusion img2img strengths on 64 held-out 1024^2 images. Table 15 reports both presence-detection and message-extraction results.

Table 15: Recovery under generative regeneration for 64 images at 1024^2. Cells report detection rate/BitAcc (%). SDXL is retested with its built-in watermark disabled; SD3.5 uses an existing evaluation of the same size. s denotes img2img strength, with 50 sampling steps and an empty prompt. CoprGuard provides only presence detection, and TrustMark-Q has no positive samples under this protocol.

	No attack	SDXL				SD3.5			
Method	Det. rate/Bit	VAE	$s = .10$	$s = .20$	$s = .25$	VAE	$s = .10$	$s = .20$	$s = .25$
RepMark	100/100	100/88	39/70	9/66	5/63	100/98	70/83	31/70	17/67
RivaGAN	100/99	89/91	9/58	0/54	0/52	97/99	95/95	75/84	55/75
DWT–DCT	77/87	0/50	0/50	0/50	0/50	5/54	3/52	2/51	0/50
DWT–DCT–SVD	100/100	3/52	0/50	0/50	0/51	91/93	77/87	39/71	28/65
Watermark Anything	100/100	100/100	8/69	2/52	0/51	100/100	100/100	66/99	28/93
DiffusionShield	100/100	0/57	0/57	0/57	0/57	0/57	0/57	0/57	0/57
CoprGuard	100/–	0/–	0/–	0/–	0/–	0/–	0/–	0/–	0/–

After disabling the built-in SDXL watermark, RepMark retains a 100% detection rate and 88% BitAcc through the SDXL VAE round trip. Watermark Anything also retains a 100% detection rate, RivaGAN reaches 89%, and the remaining methods largely fail. Under SD3.5, RepMark, RivaGAN, and Watermark Anything retain high recovery, and DWT–DCT–SVD shows partial stability. All methods decay as img2img strength increases; output PSNR relative to the watermarked input is then approximately 24–30 dB, so the result reflects both watermark removal and content change. In a new 512-image clean control, every listed method has zero false positives under SDXL; under SD3.5, only DWT–DCT–SVD produces one false positive at $s = .20$. TrustMark-Q has clean controls only, with native FPR between 1.95% and 3.91%. Corresponding positive samples are unavailable, so these FPR values are not used as substitutes for robustness measurements.

Instruction-based image editing. We use BAGEL-7B-MoT (Deng et al., 2025) for two semantic edits: locally adding a red balloon and globally converting the image to watercolor while preserving subject and composition. The unified comparison contains 16 source indices at 256^2. For each index, the instruction and BAGEL seed are identical across edited-clean images, the two RepMark paths, and seven baselines. Each method retains its fixed message and native decision rule; its own detector also evaluates the edited-clean images to measure FPR. Table 16 reports the results.

Table 16: Watermark retention after BAGEL instruction-based editing on 16 paired 256^2 samples. Detected is the number passing the native detector, Bit is mean BitAcc, and AUC uses edited-clean images from the same method as negatives. Message lengths and detection semantics differ, so BitAcc alone is not used to rank methods.

		Add red balloon			Watercolor transfer		
Method	Bits	Detected	Bit	AUC	Detected	Bit	AUC
RepMark (post-processing)	16	15/16	0.773	1.000	**14/16**	0.711	**0.949**
RepMark (in-processing)	16	**16/16**	0.785	1.000	**14/16**	0.719	**0.949**
RivaGAN	32	13/16	0.838	0.992	6/16	0.691	0.877
DWT–DCT	32	0/16	0.492	0.479	0/16	0.494	0.564
DWT–DCT–SVD	32	8/16	0.732	0.939	1/16	0.510	0.473
TrustMark-Q	61	6/16	0.898	1.000	0/16	0.665	0.877
Watermark Anything	32	10/16	0.807	1.000	3/16	0.625	0.738
DiffusionShield	128	0/16	0.570	0.480	0/16	0.570	0.496
CoprGuard	1	0/16	–	0.527	0/16	–	0.504

Local editing has limited impact on RepMark: in-processing and post-processing retain 16/16 and 15/16 native decisions, with paired AUC 1.0. Watercolor transfer induces a much larger distribution shift (mean PSNR about 11.5 dB), yet both RepMark paths retain 14/16 decisions and AUC 0.949. RivaGAN retains 6/16 and every other baseline at most 3/16. The edited-clean FPR is 0/16 for both RepMark paths. TrustMark-Q under local editing and DWT–DCT–SVD under watercolor editing each produce one false positive in 16 clean images. Before editing, all methods detect 16/16 watermarks except DWT–DCT at 15/16, so the observed degradation mainly follows editing rather than initial embedding failure.

Figure 16 shows representative paired results for in-processing. Each panel contains, from left to right, the clean input, edited clean output, watermarked input, and edited watermarked output. The instructions change the intended content, with similar semantic changes in the clean and watermarked branches.



(a) Locally adding a red balloon. (b) Global watercolor style transfer.

Figure 16: Paired examples before and after BAGEL editing using RepMark's in-processing path.

A separate RepMark-only pilot on 16 pairs at 512^2 shows the same trend. Both paths retain 16/16 detections after local addition, with BitAcc 0.992/0.984, and 10/16 after watercolor transfer, with BitAcc 0.801/0.809 and paired AUROC 0.945 for both. The studies use different resolutions and source indices, so their statistics are not combined. These remain 16-pair supplementary studies: the Wilson 95% interval for 14/16 detections is $[0.640, 0.965]$, limiting the conclusion to strong retention under the local edit and a relative advantage under global style transfer.

C.5 Writing Consistency and Shared Writing Coordinates

We construct two component controls around the complete model. *w/o consistent writing* replaces the message-only writer with an attention writer that reads both the content latent and the message, making the displacement content dependent. *w/o shared writing coordinates* keeps the fixed-message writer but uses the LoRA encoder for the first encoding, coupling writing and verification in the same private coordinates. Complete RepMark uses a message-independent writer, establishes shared writing coordinates through the frozen original FLUX.2-dev encoder, and maps attacked images into the private verification space with the LoRA encoder. Other training and evaluation settings are held fixed apart from the controlled design variable.

Table 17: Component ablation of writing consistency and shared writing coordinates at 256^2. Direct/Proxy Bit is computed on 128 images; Student Bit, AUROC, TPR, and Primary use novel generations after PixArt fine-tuning.

Method	PSNR	Direct Bit	Proxy Bit	Student Bit	AUROC	TPR@1%FPR	Primary
w/o consistent writing	**37.03**	**1.000**	0.628	0.511	0.475	0.025	0.000
w/o shared writing coordinates	31.67	0.832	0.797	0.686	0.963	0.825	0.300
RepMark (complete)	34.30	0.998	**0.837**	**0.823**	**1.000**	**1.000**	**0.900**

All three downstream experiments start from the pretrained PixArt model specified in Appendix B and use a fixed message, eight passes over the training set, 20 sampling steps, and CFG 4.5. To prevent native-threshold differences from affecting the conclusion, we compute AUROC uniformly from per-image positive and negative scores and set TPR@1%FPR using the 99th percentile of clean scores. All three have zero clean FPR under their respective native thresholds.

Without consistent writing, BitAcc is nearly perfect on the direct path but falls to 0.628 after an SDXL-VAE round trip and to 0.511 on student generations, with AUROC 0.475. Cross-sample-consistent writing raises the no-shared-writing control to 0.797 proxy-VAE BitAcc, 0.686 student BitAcc, 0.963 AUROC, and 0.825 TPR. Direct extraction alone is therefore not equivalent to model-level radioactivity, and consistent writing supports a more accumulable bias.

With the fixed-message writer held in place, removing shared writing coordinates reduces Primary to 0.300. Complete RepMark writes with the original encoder E_0 and verifies with the LoRA encoder E_ϕ, reaching 0.900 Primary, 1.000 AUROC, and 1.000 TPR while improving PSNR from 31.67 to 34.30 dB. This controlled result shows that separating cross-sample shared writing coordinates from private verification coordinates improves preservation across a VAE and propagation into the student output distribution.

C.6 Ablation of Proxy-VAE Training

For rapid controlled validation of the proxy-VAE training path, we use a simplified configuration without image-quality constraints. It uses a message-only writer, the original FLUX.2-dev encoder for writing, and a LoRA encoder for reading. When proxy-VAE training is enabled, each matched positive–negative pair undergoes an SDXL-VAE round trip with probability 0.2. Table 18 compares matched configurations without and with proxy-VAE training. Direct and SDXL-VAE metrics use the same 128-image protocol at 256^2; each configuration also supplies 500 protected images used to fine-tune a separate PixArt-Σ student for eight passes over the training set, followed by evaluation on 40 student outputs and 40 clean controls.

Table 18: Ablation of the proxy-VAE training path using a simplified configuration without image-quality constraints. VAE metrics are measured after an SDXL-VAE round trip; Student metrics use the same PixArt fine-tuning protocol.

Setting	PSNR	Direct Bit	VAE Bit	VAE Det. rate	Student Bit	Student AUROC	Student Det. rate	Student Primary
w/o proxy VAE	17.89	**0.995**	0.822	0.945	0.809	0.960	0.350	0.350
with proxy VAE	**18.00**	0.955	**0.890**	**1.000**	**0.852**	**1.000**	**0.975**	**0.900**

Enabling the proxy-VAE path raises SDXL-VAE BitAcc from 0.822 to 0.890. On downstream PixArt outputs, BitAcc, AUROC, detection rate, and Primary rise from 0.809, 0.960, 0.350, and 0.350 to 0.852, 1.000, 0.975, and 0.900, respectively. The two configurations have similar embedding PSNR, so stronger pixel distortion does not directly explain the improvement. Across six matched evaluations without proxy-VAE training, each paired with a separately trained PixArt student, mean VAE BitAcc, detection rate, and Primary are 0.812, 0.520, and 0.517 under the same 128-image protocol. Across seven matched evaluations with proxy-VAE training, each likewise paired with a separate student, the corresponding means are 0.902, 1.000, and 0.993, while mean Direct BitAcc decreases slightly from 0.998 to 0.967. These results support the role of the proxy-VAE path in cross-representation recovery and downstream propagation.

Because this rapid-validation configuration omits image-quality constraints, its embedding PSNR is only about 18 dB and is not representative of the final RepMark image quality or absolute performance. We use this controlled comparison only to assess the effect of enabling the proxy-VAE training path.