Semantic Watermarking Reinvented: Enhancing Robustness and Generation Quality with Fourier Integrity

Supplementary Material

This supplementary document provides additional context, experiments, and analyses to complement the main paper. Sec. 7 clarifies the task setup and addresses points of potential misunderstanding. Sec. 8 presents additional experimental evidence that reinforces our claims, including new results added in response to reviewer feedback. Sec. 9 provides supplementary quantitative results and extended evaluations with alternative metrics. Sec. 10 concludes with a discussion on real-world deployment considerations, highlighting the compatibility of our methods with efficient AI accelerators such as Neural Processing Units (NPUs).

7. Clarifications and Task Overview

7.1. Scope Clarification on Tampering Robustness

Our method is designed for robust watermarking. It aims to preserve the embedded information even when the content undergoes typical, non-malicious changes during distribution or transformation. It is not intended to detect peripheral tampering, which presents a fundamentally different challenge outside the scope of this work. Such tampering detection requires an alternative threat model and design considerations, often involving explicit modeling of adversarial behavior. We clarify this distinction to prevent misunderstanding regarding the intended threat model and design goals of our approach.

7.2. Taxonomy of Watermarking Methods

This section provides a brief overview of the terminology used to categorize watermarking methods discussed in the main paper. These categorizations help clarify the design characteristics of each method and contextualize the experimental results.

We group watermarking methods along the following three axes:

- **Message Type:** Methods are either *bitstream-based*, which embed and recover discrete bit sequences, or *pattern-based*, where detection relies on matching structured watermark patterns.
- **Embedding Strategy:** *Post-hoc-based* methods embed watermarks after image generation. In contrast, *merged-in-generation* methods integrate watermarking into the image synthesis process, typically within diffusion-based models.
- **Method Family:** We use the terms *classical vision* for signal processing techniques, and *deep learning-based* for methods involving trainable models. *Semantic watermarking* refers to recent approaches that embed information into the image's semantic content, often in the latent space of generative models. Semantic methods are particularly designed to be robust against semantic-preserving transformations such as regeneration, compression, or cropping.

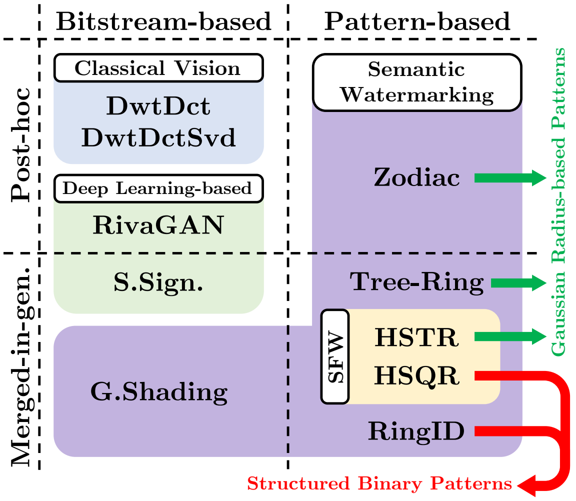

Figure 7. Taxonomy of watermarking methods evaluated in this work, categorized along three dimensions.

Among the semantic watermarking methods, we further differentiate the structure of their watermark patterns. Specifically, *Gaussian radius-based patterns*, such as Tree-Ring and HSTR, use radial embeddings with Gaussian-sampled values, whereas *structured binary patterns*, such as RingID and HSQR, resemble geometric encodings of bitstreams. Although not all of these terms are explicitly mentioned in the main paper, we include them here to help clarify the conceptual distinctions among recent semantic watermarking methods.

Fig. 7 visually summarizes the classification of all methods evaluated, including the baseline that will be introduced in Sec. 8.2.

7.3. Identification Protocol for Tree-Ring

This section outlines the identification procedure used for the Tree-Ring baseline. Following the multi-key evaluation protocol introduced in RingID [11], we construct a candidate key pool for each target capacity. Specifically, we generate a large set of key embeddings corresponding to different watermark messages. During evaluation, the extracted

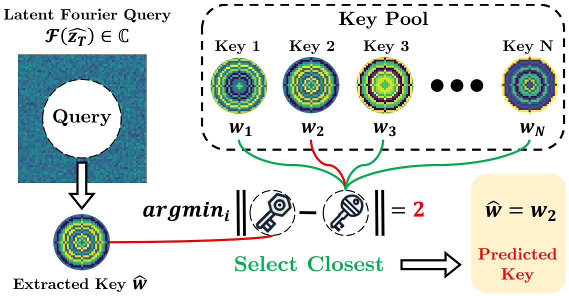

Figure 8. Schematic illustration of the adapted identification protocol for Tree-Ring. Multiple candidate keys are generated and the extracted watermark is matched to the closest key based on L_1 distance.

pattern from a watermarked image is matched against all keys in the pool using L_1 distance in the embedding space, and the message associated with the closest key is selected as the predicted output.

Fig. 8 provides a schematic overview of this process, illustrating how multiple keys are generated and compared in the identification pipeline. This procedure enables Tree-Ring to be evaluated under the same capacity-controlled setting as other semantic methods. This protocol is also applied to other Gaussian radius-based methods, including Zodiac and the proposed HSTR.

8. Additional Experimental Evidence

8.1. Processing Time and Detection Performance

This section presents the processing time and detection performance (verification and identification) of different watermarking methods. As shown in Tab. 5, the *merged-in-generation* approach does not introduce additional processing time since the watermarking process is inherently integrated into the diffusion-based image generation. The proposed method achieves superior detection performance without requiring additional processing time, demonstrating its efficiency. On the other hand, *post-hoc-based* approaches require per-image processing time, with some methods incurring significant computational costs. In particular, Zodiac demands several minutes per image as it requires multiple rounds of diffusion-based generation and latent vector optimization iterations for embedding the semantic watermark pattern, resulting in excessive computational overhead.

8.2. Performance of Gaussian Shading as a Baseline

Following feedback received during review, this section presents the performance of Gaussian Shading (G.Shading) on the MS-COCO dataset. Tab. 6 shows the detection results under the same attack settings as those used in Tab. 1 and Tab. 2 of the main paper. Since G.Shading is evaluated as a bitstream-based setting, we report Bit Accuracy for verification and Perfect Match Rate for identification as the detection metrics. In addition to detection performance, we also evaluate the generative quality as a supplement to Tab. 3. G.Shading achieves an FID of 24.778 and a CLIP score of 0.330, which are comparable to those of our method HSQR (FID: 24.895, CLIP: 0.330).

Table 5. Evaluation of watermarking methods based on processing time, verification, and identification performance. The best performance for each item is highlighted with shading, while bold text specifically marks the excessive processing time in Zodiac. *Vrf.* and *Idf.* denote the average detection performance in verification and identification tasks, respectively.

Methods		Processing Time ↓	*Vrf.*	*Idf.*
Post-hoc Based	DwtDct	0.03 (s/img)	0.637	0.083
	DwtDctSvd	0.07 (s/img)	0.742	0.258
	RivaGAN	0.41 (s/img)	0.857	0.482
	Zodiac	**7.36 (m/img)**	0.962	0.000
Merged in Generation	S.Sign.	0.00 (s/img)	0.836	0.265
	Tree-Ring	0.00 (s/img)	0.655	0.114
	RingID	0.00 (s/img)	0.997	0.964
	HSTR (ours)	0.00 (s/img)	0.971	0.889
	HSQR (ours)	0.00 (s/img)	0.997	0.985

Table 6. Verification and identification performance of Gaussian Shading under the same attack settings as in Tab. 1 and Tab. 2 of the main paper (MS-COCO only).

Attack Type	*Verification*			*Identification*		
	G.Shading	HSTR	HSQR	G.Shading	HSTR	HSQR
No Attack	1.000	1.000	1.000	1.000	1.000	1.000
Bright.	0.962	0.899	0.991	0.522	0.714	0.958
Cont.	1.000	1.000	1.000	0.998	0.999	1.000
JPEG	0.992	0.994	1.000	0.724	0.886	0.994
Blur	1.000	1.000	1.000	0.999	0.998	1.000
Noise	0.997	0.806	0.983	0.919	0.460	0.901
BM3D	0.999	0.999	1.000	0.926	0.972	0.999
VAE-B	0.982	0.973	0.992	0.636	0.833	0.980
VAE-C	0.987	0.982	1.000	0.657	0.831	0.987
Diff.	0.999	0.997	1.000	0.827	0.971	0.999
C.C.	0.998	1.000	1.000	0.658	1.000	1.000
R.C.	1.000	1.000	1.000	0.986	1.000	1.000
Avg	0.993	0.971	0.997	0.821	0.889	0.985

Table 7. Normality assessment of latent distributions (1,000 samples). HSTR better preserves Gaussianity than Tree-Ring, as shown by standard deviation, KS p-value, and failure rate.

Methods	Mean	Std. Dev.	KS p-value ↑	KS failure rate ↓
Tree-Ring	0.0004	0.9620	0.2404	0.234
HSTR (ours)	-0.0003	1.0000	0.4227	0.071

8.3. Preservation of Gaussianity

We assess the normality of 1,000 latent samples, as reported in Tab. 7. Compared to Tree-Ring, HSTR more closely aligns with $\mathcal{N}(0, 1)$ in terms of standard deviation and the

Table 8. Ablation study on the impact of Hermitian SFW and center-aware embedding in terms of robustness and generative quality.

Case	SFW	Center	*Signal.*	*Regen.*	*Crop.*	Avg	FID ↓	CLIP ↑
A	×	×	0.136	0.070	0.021	0.114	26.418	0.326
B	✓	×	0.856	0.812	0.374	0.777	25.071	0.329
C	✓	✓	0.838	0.878	1.000	0.889	25.062	0.329

Table 9. Average PSNR, SSIM, and LPIPS values for each of the 11 attack types, computed over 1,000 MS-COCO generated images. These values reflect the typical level of distortion introduced by each attack.

Attack Type	PSNR ↑	SSIM ↑	LPIPS ↓
Bright.	28.421	0.558	0.383
Cont.	28.015	0.824	0.092
JPEG	32.909	0.898	0.066
Blur	34.419	0.902	0.023
Noise	44.926	0.894	0.113
BM3D	35.648	0.910	0.074
VAE-B	33.594	0.884	0.093
VAE-C	33.950	0.896	0.083
Diff.	31.224	0.795	0.109
C.C.	30.959	0.503	0.431
R.C.	33.164	0.702	0.298

Kolmogorov–Smirnov (KS) test. This includes higher p-values and lower failure rates, indicating stronger statistical consistency.

8.4. Disentangling the Contributions

We conduct an ablation analysis to examine the individual contributions of each component, with the results presented in Tab. 8. From Tree-Ring (A), applying SFW (B) improves frequency integrity, which enhances robustness to signal and regeneration attacks as well as generative quality. Adding center-aware embedding (C), which corresponds to the proposed HSTR method, significantly enhances robustness to cropping attacks. Note that *Signal.*, *Regen.*, and *Crop.* denote the average identification accuracy across the respective attack types. These results suggest that both components are necessary and complementary.

8.5. Post-Attack Image Quality Assessment

To complement the robustness evaluation, we provide an assessment of image quality degradation caused by various attacks. Tab. 9 reports the average PSNR, SSIM, and LPIPS values computed over 1,000 MS-COCO generated images after applying each of the 11 attack types used in the main paper. We report PSNR and SSIM to measure pixel-level and structural similarity respectively, and include LPIPS to capture perceptual quality more closely aligned with human judgment. These metrics help ensure that attack strengths remain realistic and consistent across evaluation scenarios.

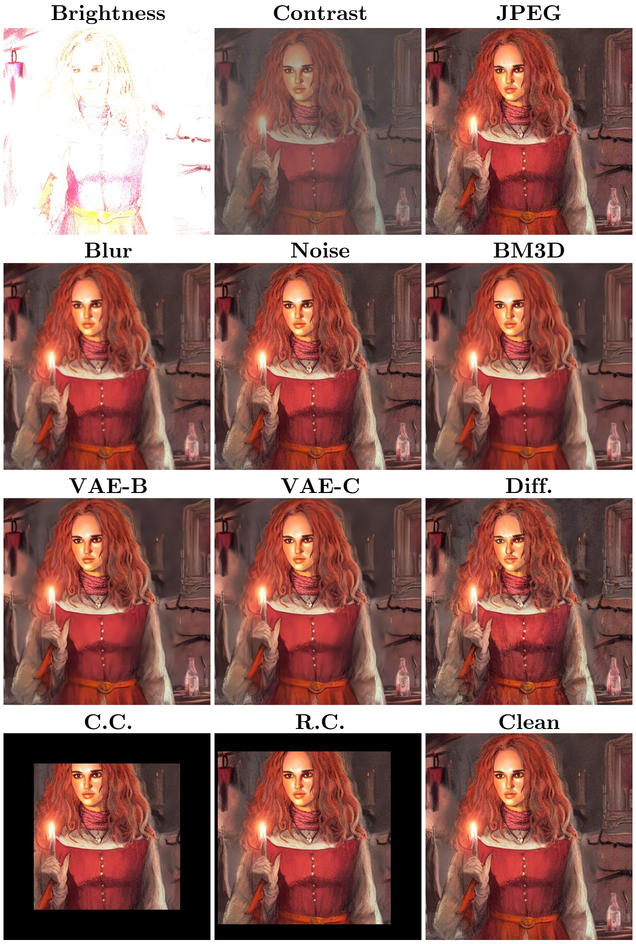

Figure 9. Visual examples of all 11 attacks applied to a single clean image. The figure illustrates the perceptual effects of each attack type relative to the original input.

In addition, Fig. 9 visualizes the effect of all 11 attacks on a single clean image, illustrating the diverse perceptual degradation introduced by each attack. Notably, the three examples in the third row of the figure, corresponding to regeneration attacks, appear visually high-quality from a classical signal processing perspective. Despite the minimal perceptual degradation, many baseline methods fail to maintain correct detection under these attacks, as shown in Tab. 1 (verification performance). This suggests that the attack strength is not weak, even if visual quality remains high. It also highlights that improving robustness to regeneration attacks remains a critical challenge, both for our method and for future research in this area.

Table 10. Verification and identification performance of semantic watermarking methods under diffusion-based regeneration attacks with varying noise steps. The results are based on 1,000 images generated from the MS-COCO dataset. Larger noise steps indicate stronger attack strength.

Task	Methods	Noise Step				
		20	60	100	140	180
Vrf.	Tree-Ring	0.701	0.543	0.404	0.317	0.262
	HSTR (ours)	1.000	0.997	0.998	0.987	0.980
	RingID	1.000	1.000	1.000	1.000	1.000
	HSQR (ours)	1.000	1.000	1.000	1.000	1.000
Idf.	Tree-Ring	0.144	0.054	0.034	0.014	0.012
	HSTR (ours)	0.999	0.971	0.926	0.868	0.781
	RingID	1.000	0.998	0.995	0.993	0.990
	HSQR (ours)	0.999	0.999	0.999	0.998	0.997

8.6. Ablation on Regeneration Strength

To further investigate the robustness of our method against regeneration attacks, we conduct an ablation study by varying the noise strength in the diffusion-based regeneration attack. This experiment addresses the concern that the regeneration attacks used in the main paper may have been too weak compared to pixel-level Gaussian noise attacks, a typical signal processing perturbation.

Following the setup of Zhao et al. [56], the attack applies additive noise in the latent space using a formulation similar to the forward process in DDPM [25]:

$$z_{t^*} \leftarrow \sqrt{\alpha(t^*)}z_0 + \sqrt{1-\alpha(t^*)}\epsilon$$

Here, z_0 denotes the encoded latent representation of the image, and ϵ is standard Gaussian noise. The variable t^* indicates the noise step that controls attack strength. The attacked image is then regenerated through the denoising process.

Tab. 10 reports the verification and identification performance of semantic watermarking methods that follow the *merged-in-generation* scheme, evaluated under varying attack steps $t^* \in \{20, 60, 100, 140, 180\}$. Following the main paper, verification is measured by TPR@1%FPR and identification by accuracy, reported as Perfect Match Rate. Detection performance progressively decreases as the noise step increases, indicating a corresponding increase in attack strength. A higher noise step results in lower detection performance, suggesting a stronger perturbation effect. The proposed HSQR achieves the most robust detection performance under these stronger attack levels. Focusing on Gaussian radius-based methods (Tree-Ring and HSTR), we observe that the proposed HSTR demonstrates significantly improved robustness over Tree-Ring. This highlights the effectiveness of the Hermitian SFW component under challenging regeneration scenarios.

9. Supplementary Experimental Results

9.1. Reporting Bit Accuracy Results

In the main paper, we use Bit Accuracy for verification and Perfect Match Rate for identification to evaluate the detection performance of bitstream-based approaches in Sec. 5.2. Following feedback received during review, we acknowledge that Bit Accuracy offers a more fine-grained perspective, particularly in high-capacity or multi-user settings. To complement the original results, we report unified detection performance in terms of Bit Accuracy across all methods in Tab. 11. Here, we adopt a strict evaluation criterion for the semantic methods: if the predicted pattern does not exactly match the ground-truth pattern, the Bit Accuracy for that sample is set to zero. We believe these results provide a more comprehensive view of overall detection performance.

9.2. Further Results for Verification

This section provides supplementary results for semantic methods on the verification task introduced in Sec. 5.2.

- Fig. 10, Fig. 11, and Fig. 12 illustrate the Receiver Operating Characteristic (ROC) curves for different datasets under various attack scenarios.
- Tab. 12 and Tab. 13 summarize the corresponding Area Under the Curve (AUC) values and maximum accuracy for each dataset.

9.3. Numerical Results for Ablation Study

This section presents the numerical data corresponding to the figures in Sec. 5.3 (Ablation Study).

- Tab. 14 provides detailed results for the ablation study on Hermitian SFW cases presented in Tab. 4.
- Tab. 15 shows the identification accuracy under center crop and random crop attacks at different crop scales, corresponding to Fig. 5.
- Tab. 16 presents the average identification accuracy across clean conditions and 11 attack scenarios for different watermarking capacities, as shown in Fig. 6. For each capacity, we report its associated embedding density, computed based on a fixed image resolution of 512×512. This supplements the capacity-related analysis by making the notion of embedding density explicit, as suggested during the review.

9.4. Qualitative Analysis for Semantic Methods

This section presents qualitative results for semantic watermarking methods following the *merged-in-generation* scheme, showcasing generated images from the same prompt, as illustrated in Fig. 13. In the case of RingID, excessive emphasis on detection performance at the expense of image quality results in imbalanced trade-offs, causing noticeable *ring-like* artifacts in the generated images. This phenomenon aligns with the low CLIP score

Table 11. Unified detection performance reported in terms of Bit Accuracy for all methods, including re-evaluation of semantic watermarks.

Datasets	Methods	No Attack	Signal Processing Attack						Regeneration Attack			Cropping Attack		Avg
		Clean	Bright.	Cont.	JPEG	Blur	Noise	BM3D	VAE-B	VAE-C	Diff.	C.C.	R.C.	
MS-COCO	DwtDct	0.863	0.572	0.522	0.516	0.677	0.859	0.532	0.523	0.521	0.519	0.729	0.810	0.637
	DwtDctSvd	1.000	0.555	0.473	0.602	1.000	1.000	0.784	0.648	0.596	0.644	0.744	0.861	0.742
	RivaGAN	0.999	0.862	0.986	0.821	0.998	0.969	0.934	0.570	0.552	0.608	0.991	0.995	0.857
	S.Sign.	0.995	0.894	0.978	0.806	0.911	0.721	0.838	0.717	0.715	0.478	0.987	0.991	0.836
	Tree-Ring	0.303	0.087	0.207	0.072	0.256	0.030	0.162	0.083	0.072	0.054	0.009	0.033	0.114
	Zodiac	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	HSTR (ours)	1.000	0.714	0.999	0.886	0.998	0.460	0.972	0.833	0.831	0.971	1.000	1.000	0.889
	RingID	1.000	0.875	1.000	0.975	1.000	0.919	0.996	0.978	0.970	0.998	0.874	0.978	0.964
	HSQR (ours)	1.000	0.958	1.000	0.994	1.000	0.901	0.999	0.980	0.987	0.999	1.000	1.000	0.985
SD-Prompts	DwtDct	0.819	0.557	0.516	0.506	0.685	0.822	0.530	0.513	0.512	0.509	0.723	0.794	0.624
	DwtDctSvd	1.000	0.537	0.459	0.610	0.999	0.998	0.859	0.659	0.620	0.623	0.743	0.860	0.747
	RivaGAN	0.991	0.823	0.963	0.810	0.988	0.961	0.915	0.572	0.535	0.567	0.980	0.983	0.841
	S.Sign.	0.994	0.899	0.967	0.769	0.888	0.742	0.809	0.677	0.671	0.493	0.983	0.990	0.824
	Tree-Ring	0.288	0.094	0.189	0.051	0.235	0.034	0.159	0.079	0.076	0.056	0.012	0.041	0.110
	Zodiac	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	HSTR (ours)	1.000	0.655	0.999	0.863	0.999	0.555	0.980	0.846	0.847	0.973	1.000	1.000	0.893
	RingID	1.000	0.885	1.000	0.976	0.998	0.886	0.993	0.980	0.973	0.995	0.876	0.981	0.962
	HSQR (ours)	1.000	0.930	1.000	0.994	1.000	0.942	0.999	0.991	0.997	1.000	1.000	1.000	0.988
DiffusionDB	DwtDct	0.842	0.563	0.515	0.509	0.672	0.829	0.526	0.513	0.514	0.512	0.723	0.801	0.627
	DwtDctSvd	0.998	0.558	0.463	0.593	0.997	0.995	0.830	0.658	0.608	0.621	0.742	0.860	0.744
	RivaGAN	0.987	0.839	0.960	0.790	0.985	0.937	0.893	0.553	0.518	0.556	0.974	0.979	0.831
	S.Sign.	0.990	0.890	0.967	0.787	0.889	0.726	0.819	0.690	0.687	0.496	0.981	0.986	0.826
	Tree-Ring	0.280	0.095	0.190	0.059	0.233	0.037	0.145	0.081	0.072	0.050	0.013	0.039	0.108
	Zodiac	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	HSTR (ours)	0.996	0.721	0.992	0.854	0.989	0.563	0.958	0.830	0.821	0.952	0.996	0.996	0.889
	RingID	1.000	0.895	1.000	0.947	0.996	0.871	0.992	0.968	0.958	0.990	0.875	0.984	0.956
	HSQR (ours)	1.000	0.954	1.000	0.988	1.000	0.906	0.998	0.982	0.991	0.994	1.000	1.000	0.984

Table 12. AUC values of semantic methods for verification, evaluated across different datasets and attack scenarios.

Datasets	Methods	No Attack	Signal Processing Attack						Regeneration Attack			Cropping Attack		Avg
		Clean	Bright.	Cont.	JPEG	Blur	Noise	BM3D	VAE-B	VAE-C	Diff.	C.C.	R.C.	
MS-COCO	Tree-Ring	0.997	0.895	0.990	0.923	0.994	0.870	0.977	0.912	0.924	0.921	0.913	0.962	0.940
	Zodiac	1.000	0.978	1.000	0.998	1.000	0.978	1.000	0.989	0.996	0.997	0.999	1.000	0.995
	HSTR (ours)	1.000	0.992	1.000	1.000	1.000	0.986	1.000	0.995	0.999	1.000	1.000	1.000	0.998
	RingID	1.000	0.999	1.000	1.000	1.000	0.998	1.000	0.994	1.000	1.000	1.000	1.000	0.999
	HSQR (ours)	1.000	0.999	1.000	1.000	1.000	0.996	1.000	0.997	1.000	1.000	1.000	1.000	0.999
SD-Prompts	Tree-Ring	0.995	0.892	0.985	0.911	0.991	0.875	0.980	0.915	0.928	0.911	0.906	0.957	0.937
	Zodiac	1.000	0.940	1.000	0.998	1.000	0.977	1.000	0.985	0.998	0.995	1.000	1.000	0.991
	HSTR (ours)	1.000	0.979	1.000	1.000	1.000	0.986	1.000	0.996	0.999	1.000	1.000	1.000	0.997
	RingID	1.000	0.997	1.000	1.000	1.000	0.999	1.000	0.998	1.000	1.000	1.000	1.000	1.000
	HSQR (ours)	1.000	0.997	1.000	1.000	1.000	0.997	1.000	0.999	1.000	1.000	1.000	1.000	0.999
DiffusionDB	Tree-Ring	0.993	0.894	0.983	0.902	0.988	0.856	0.971	0.905	0.912	0.900	0.904	0.955	0.930
	Zodiac	0.997	0.958	0.997	0.990	0.997	0.942	0.996	0.975	0.984	0.983	0.993	0.997	0.984
	HSTR (ours)	1.000	0.988	1.000	0.999	1.000	0.974	1.000	0.995	0.998	1.000	1.000	1.000	0.996
	RingID	1.000	0.999	1.000	1.000	1.000	0.994	1.000	0.997	1.000	1.000	1.000	1.000	0.999
	HSQR (ours)	1.000	0.997	1.000	1.000	1.000	0.994	1.000	0.999	1.000	1.000	1.000	1.000	0.999

(0.324) reported in Tab. 3, indicating degraded text-image alignment. Furthermore, for Tree-Ring, which employs a Gaussian radius-based watermark pattern, the proposed method HSTR preserves Fourier integrity while embedding the same pattern. As a result, HSTR improves the CLIP score from 0.326 to 0.329, demonstrating its ability to enhance the quality of diffusion-generated images.

To further assess perceptual quality, we conduct a Mean Opinion Score (MOS) study based on human evaluations. For each of 10 prompts, we present the corresponding images generated by Tree-Ring, RingID, HSTR, and HSQR as a group, and ask 10 human evaluators to rate each image individually on a scale from 1 (very poor) to 5 (excellent). Participants assign a separate score to each image based on its visual quality. The resulting average MOS scores are 2.82 for RingID, 3.54 for Tree-Ring, 3.69 for

Table 13. Maximum verification accuracy for semantic methods across different datasets and attack scenarios.

Datasets	Methods	No Attack	Signal Processing Attack						Regeneration Attack			Cropping Attack		Avg
		Clean	Bright.	Cont.	JPEG	Blur	Noise	BM3D	VAE-B	VAE-C	Diff.	C.C.	R.C.	
MS-COCO	Tree-Ring	0.979	0.828	0.959	0.852	0.968	0.808	0.930	0.846	0.856	0.849	0.850	0.911	0.886
	Zodiac	0.998	0.931	0.998	0.984	0.998	0.941	0.998	0.968	0.977	0.983	0.992	0.997	0.980
	HSTR (ours)	1.000	0.963	1.000	0.993	1.000	0.943	0.999	0.982	0.988	0.997	1.000	1.000	0.989
	RingID	1.000	0.991	1.000	1.000	1.000	0.990	1.000	0.996	1.000	1.000	1.000	1.000	0.998
	HSQR (ours)	1.000	0.992	1.000	1.000	1.000	0.988	1.000	0.996	1.000	1.000	1.000	1.000	0.998
SD-Prompts	Tree-Ring	0.973	0.822	0.951	0.835	0.962	0.805	0.933	0.843	0.857	0.838	0.847	0.898	0.880
	Zodiac	0.998	0.888	0.999	0.986	0.999	0.954	0.998	0.967	0.983	0.978	0.994	0.998	0.978
	HSTR (ours)	1.000	0.939	1.000	0.992	1.000	0.945	0.998	0.989	0.989	0.998	1.000	1.000	0.987
	RingID	1.000	0.984	1.000	1.000	1.000	0.991	1.000	0.998	1.000	1.000	1.000	1.000	0.998
	HSQR (ours)	1.000	0.980	1.000	0.999	1.000	0.993	1.000	0.998	1.000	1.000	1.000	1.000	0.997
DiffusionDB	Tree-Ring	0.968	0.823	0.944	0.833	0.952	0.795	0.919	0.839	0.844	0.827	0.843	0.897	0.874
	Zodiac	0.993	0.903	0.991	0.964	0.992	0.920	0.989	0.951	0.964	0.952	0.981	0.991	0.966
	HSTR (ours)	0.999	0.951	0.998	0.988	0.997	0.917	0.994	0.980	0.982	0.992	0.999	0.999	0.983
	RingID	1.000	0.991	1.000	0.999	1.000	0.978	1.000	0.998	1.000	1.000	1.000	1.000	0.997
	HSQR (ours)	1.000	0.987	1.000	0.999	1.000	0.983	1.000	0.997	0.999	1.000	1.000	1.000	0.997

Table 14. Detailed verification and identification performance for Hermitian SFW ablation cases, supplementing Tab. 4 in the main paper.

Task	Case	No Attack	Signal Processing Attack						Regeneration Attack			Cropping Attack		Avg
		Clean	Bright.	Cont.	JPEG	Blur	Noise	BM3D	VAE-B	VAE-C	Diff.	C.C.	R.C.	
Vrf.	A	0.957	0.452	0.900	0.548	0.934	0.412	0.815	0.501	0.536	0.509	0.734	0.543	0.653
	B	1.000	0.601	1.000	0.772	1.000	0.588	0.977	0.737	0.774	0.550	0.853	0.810	0.805
	C	1.000	0.769	1.000	0.975	1.000	0.627	0.991	0.931	0.920	1.000	1.000	0.990	0.936
	D	1.000	0.899	1.000	0.994	1.000	0.806	0.999	0.981	0.982	1.000	1.000	0.997	0.971
Idf.	A	0.303	0.090	0.207	0.072	0.256	0.030	0.162	0.084	0.072	0.009	0.033	0.054	0.114
	B	0.982	0.320	0.854	0.268	0.904	0.106	0.624	0.306	0.293	0.018	0.109	0.202	0.416
	C	0.997	0.505	0.984	0.687	0.980	0.212	0.837	0.631	0.613	1.000	1.000	0.852	0.775
	D	1.000	0.714	0.999	0.886	0.998	0.460	0.972	0.841	0.831	1.000	1.000	0.971	0.889

Table 15. Identification accuracy under center crop and random crop attacks at different crop scales. These results correspond to Fig. 5

Center Crop Attack

Methods	Crop Scale						
	0.2	0.3	0.4	0.5	0.6	0.7	0.8
RingID	0.153	0.369	0.647	0.874	0.934	0.974	0.992
HSTR (ours)	0.818	0.997	1.000	1.000	1.000	1.000	1.000
HSQR (ours)	0.555	0.998	1.000	1.000	1.000	1.000	1.000

Random Crop Attack

Methods	Crop Scale						
	0.2	0.3	0.4	0.5	0.6	0.7	0.8
RingID	0.496	0.559	0.774	0.919	0.971	0.970	0.997
HSTR (ours)	0.489	0.903	0.992	0.999	1.000	1.000	1.000
HSQR (ours)	0.955	0.999	0.999	1.000	1.000	1.000	1.000

HSQR, and 3.86 for HSTR. While HSTR ranked highest in MOS, HSQR remains strong across both human ratings and CLIP/FID metrics, showing consistent perceptual quality overall.

Table 16. Average identification accuracy across watermark message capacities. The values are computed over all attack scenarios for semantic methods. These results correspond to Fig. 6

Methods	Embedding Density (10^{-5} bpp)					
	2.29	3.05	3.81	4.20	4.58	4.96
Tree-Ring	0.338	0.271	0.136	0.114	0.083	0.064
Zodiac	0.027	0.000	0.000	0.000	0.000	0.000
HSTR (ours)	0.960	0.936	0.913	0.889	0.881	0.862
RingID	0.995	0.989	0.978	0.964	0.940	0.888
HSQR (ours)	0.993	0.990	0.987	0.985	0.984	0.981

10. Outlook and Deployment Considerations

The growing accessibility of LDMs has enabled an unprecedented scale of generative content creation. As synthetic media becomes ubiquitous, embedding provenance signals at generation time, rather than through costly post-processing, will become increasingly vital.

Meanwhile, modern NPUs are optimized for low-power, high-throughput AI inference. These accelerators favor low-precision formats such as FP16 or INT8, aligning well with the inference-only use of lightweight generative models like Stable Diffusion.

Our proposed watermarking methods, HSTR and HSQR, are inherently compatible with this direction. They require no additional training, integrate seamlessly into the generation pipeline, and avoid post-hoc overhead. This *merged-in-generation* design, combined with semantic robustness and compatibility with quantized LDMs, positions our approach as a strong candidate for deployment in scalable, energy-efficient environments.

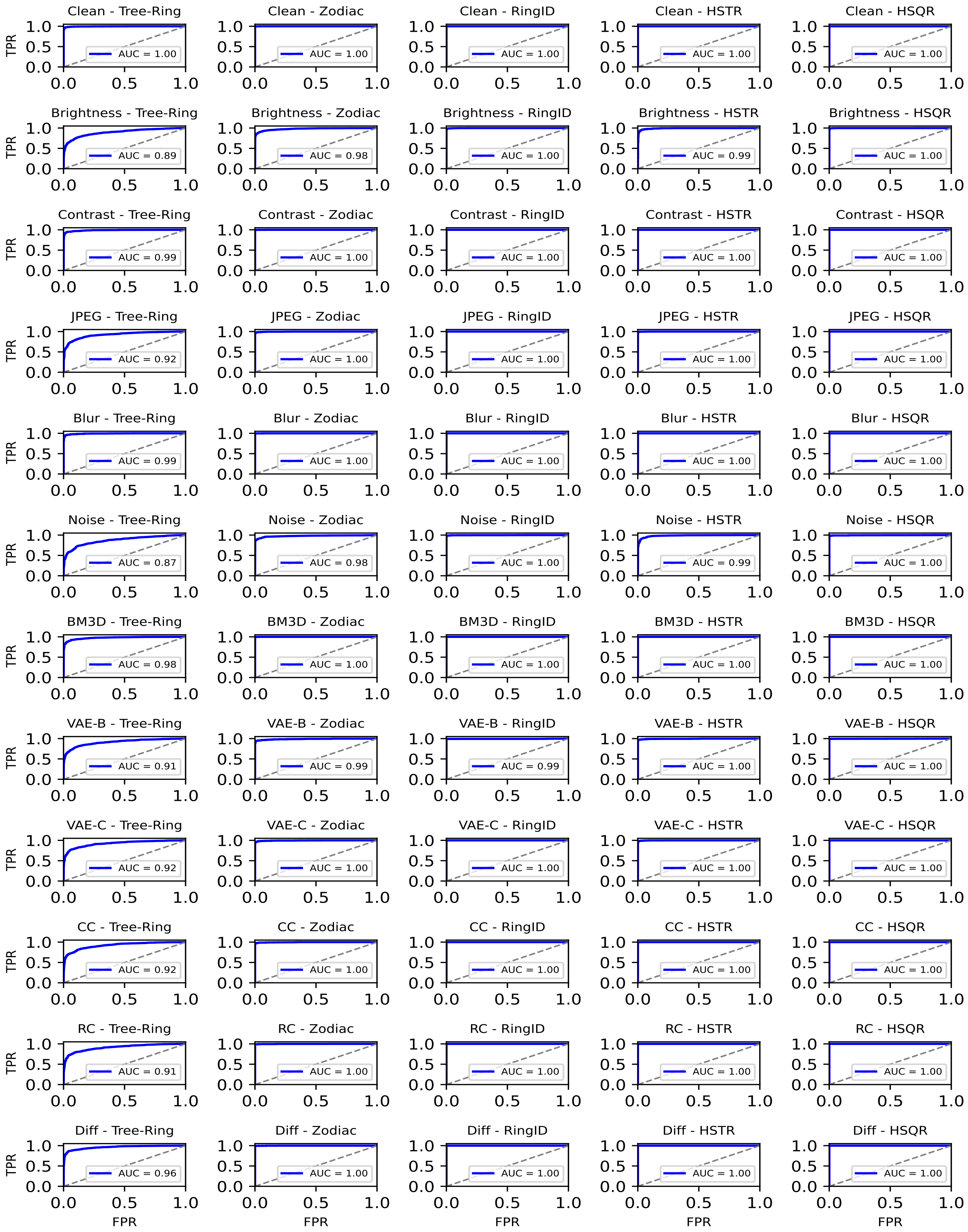

Figure 10. ROC curve for verification performance on MS-COCO under different attack scenarios.

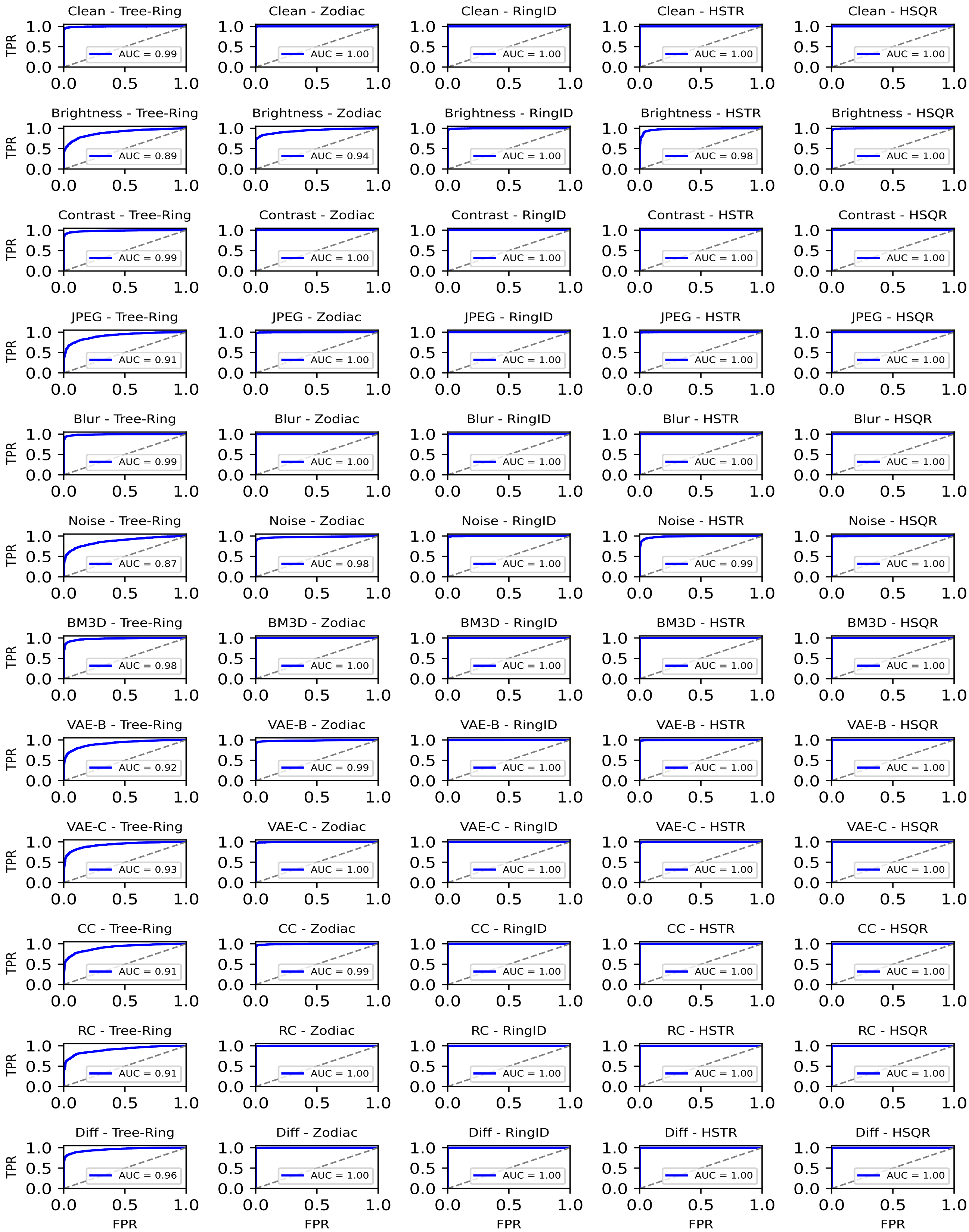

Figure 11. ROC curve for verification performance on SD-Prompts under different attack scenarios.

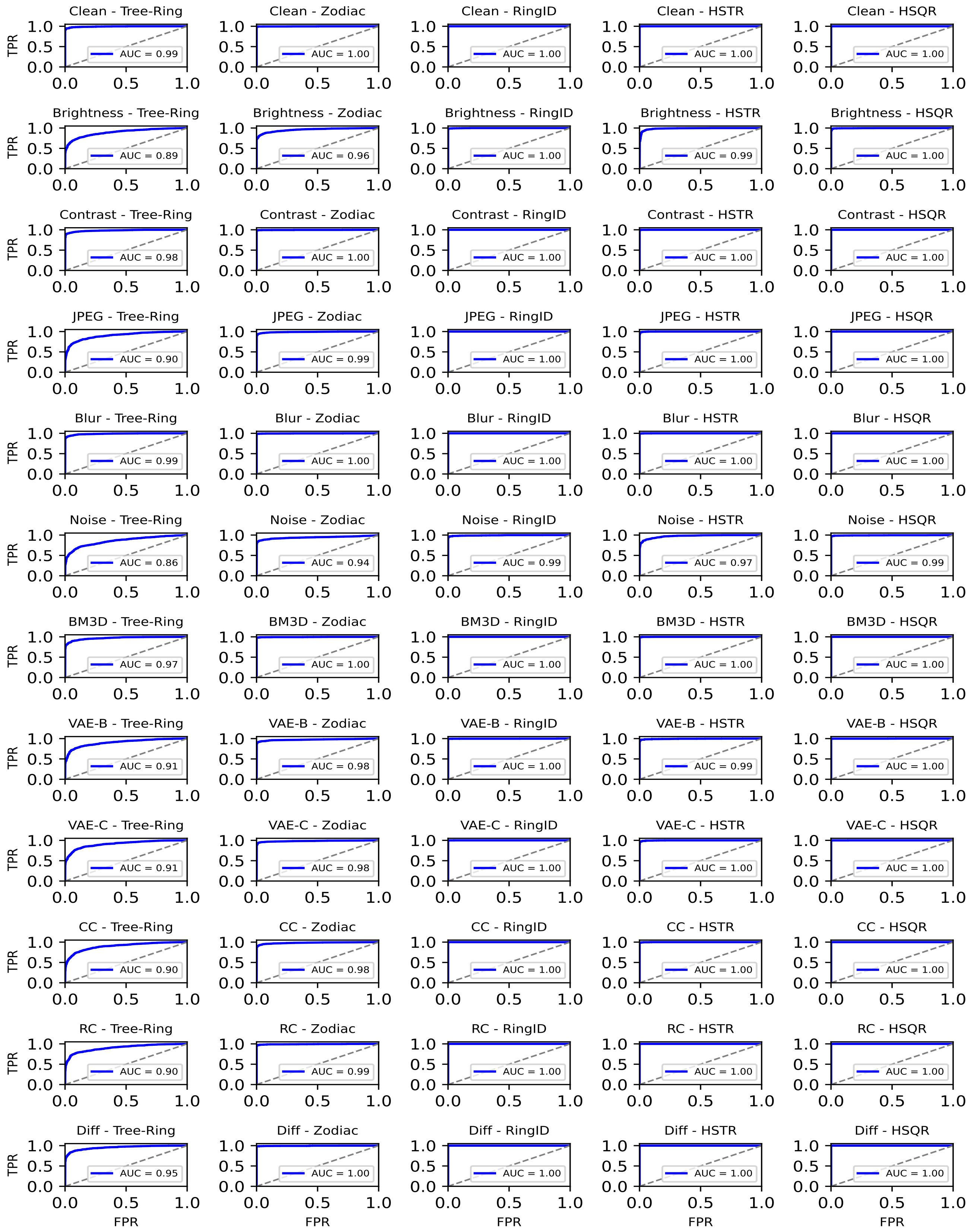

Figure 12. ROC curve for verification performance on DiffusionDB under different attack scenarios.

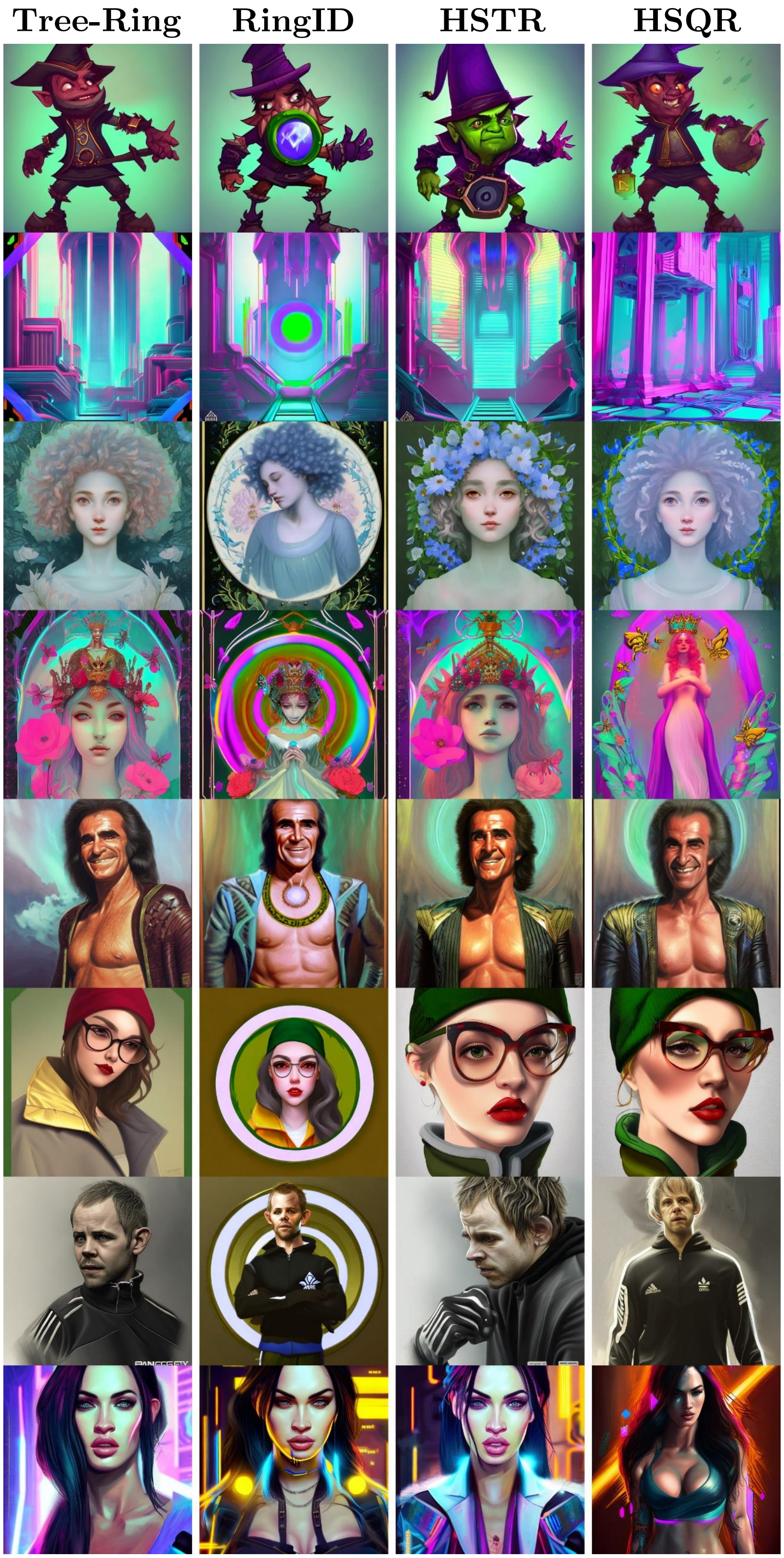

Figure 13. Qualitative comparison of semantic watermarking methods following the *merged-in-generation* scheme. The generated images are produced from the same prompt, illustrating the visual differences across different watermarking approaches.