SpecGuard: Spectral Projection-based Advanced Invisible Watermarking

Supplementary Material

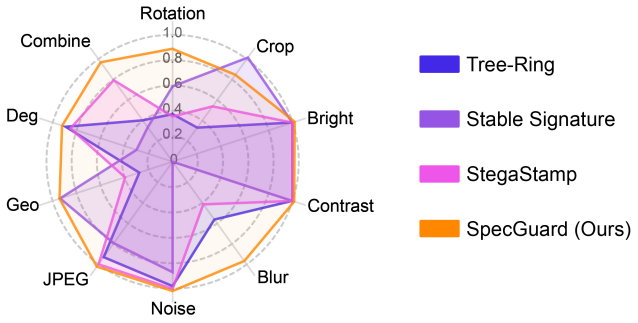

Figure 1. Comparison of SOTA watermarking methods in terms of average TPR@0.1%FPR (90% of watermarked images are correctly detected at 0.1% false positive rate) under different attacks.

1. Summary of Notations

To ensure clarity in understanding SpecGuard's mathematical formulation, we summarize the key notations used throughout the methodology (Sec. 3) of the main paper. The complete set of notations is presented in Tab. 1.

2. Impact of Parseval's Theorem in Message Extraction

To achieve robust and efficient decoding as detailed in Sec. 3.2 of the main paper, SpecGuard leverages Parseval's theorem [4], a fundamental principle in signal processing, which establishes energy equivalence between spatial and spectral domains. Formally, Parseval's theorem is defined as follows:

$$\sum_{x,y} |I(x,y)|^2 = \sum_{u,v} |\zeta(u,v)|^2, \tag{1}$$

where $I(x,y)$ denotes spatial-domain pixel intensities, and $\zeta(u,v)$ represent their corresponding spectral-domain coefficients.

In SpecGuard, watermark embedding modifies selected spectral coefficients, introducing subtle local energy variations. The embedding process employs a strength factor s, adjusting spectral energy differences as follows:

$$\zeta_{\text{embedded}}(u,v) = \zeta(u,v) + s \cdot W(u,v), \tag{2}$$

where $\zeta_{\text{embedded}}(u,v)$ denotes modified coefficients and $W(u,v)$ is the spectral-domain watermark signal. Although local energy distribution is altered, the overall signal energy remains constant as guaranteed by Parseval's theorem as follows:

$$\sum_{x,y} |I_(x,y)|^2 = \sum_{u,v} |\zeta_{\text{embedded}}(u,v)|^2. \tag{3}$$

During decoding, these local spectral energy variations, preserved due to total energy constancy, allow stable watermark extraction. Specifically, the decoder computes spectral projections via FFT approximation to isolate embedded spectral energy patterns as follows:

$$S^{\text{sp}}_{D_{HH}} = \text{SpectralProjectionFFT}(S^{\text{high}}_{D_{HH}}). \tag{4}$$

The decoder subsequently employs a dynamically optimized threshold θ to differentiate watermark signals from noise as follows:

$$D_M[i] = \begin{cases} 1 & \text{if } S^{\text{sp}}_{D_{HH}}[i] > \theta, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

The adaptive threshold θ is optimized via gradient descent during training, adapting to spectral energy distributions as follows:

$$\theta \leftarrow \theta - \eta \cdot \frac{\partial L_{\text{dec}}}{\partial \theta}, \tag{6}$$

where L_{dec} is the decoding loss, and η is the learning rate. Thus, Parseval's theorem critically supports SpecGuard by preserving total spectral energy, enabling stable differentiation of watermark bits and reliable decoding even under diverse real-world image distortions and adversarial attacks.

3. Mathematical Proof

3.1. Proof for S_{HH} Band of Wavelet Projection.

Here we presented a proof of one of the wavelet projections S_{HH} from Eq. (4) based on the Eq. (6) of the main paper.

$$\psi^D_j(u) = 2^{j/2}\psi^D(2^j u), \quad \text{//1D wavelet}$$

$$\psi^D_{j,m}(u) = 2^{j/2}\psi^D(2^j u - m), \quad \text{//Translation}$$

$$\psi^D_{j,m,n}(u,v) = 2^{j/2}\psi^D(2^j u - m) \cdot \psi^D(2^j v - n), \quad \text{//2D wavelet}$$

$$S_{HH}(j,m,n) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I(u,v) \cdot \psi^D_{j,m,n}(u,v)\, du\, dv, \quad \text{//Projection}$$

Notation	Description
I	Cover image
I_{embedded}	Watermarked image
M	Watermark message
c	Number of channels (e.g., RGB has $c = 3$)
H, W	Height and width of the image
$W(a, b)$	Wavelet transform of signal $f(x)$
a, b	Scaling and translation parameters in wavelet transform
ψ	Mother wavelet function
$d)$	Direction of each wavelet components derived from ψ
$\phi(u, v), \psi_H(u, v), \psi_V(u, v), \psi_D(u, v)$	Every directional scaling and wavelet basis components
$S_{LL}, S_{LH}, S_{HL}, S_{HH}$	Wavelet sub-bands (low and high frequency components)
β_j	Feature set capturing frequency and spatial details
κ	Decomposition level determined by image complexity
$T(x, y)$	Pixel intensity in high-frequency sub-band S_{HH}
$\zeta(u, v)$	Spectral projection coefficients
s	Strength factor controlling embedding intensity
(c_x, c_y)	Center coordinates of the image
$D(x_i, y_i)$	Euclidean distance from the center
r	Radius of embedding region
W_c	Selected watermark channel for embedding
θ	Learnable threshold for watermark extraction
$F(u, v)$	2D Fast Fourier Transform (FFT) of the extended signal
$L_{\text{enc}}, L_{\text{dec}}$	Encoder and decoder loss functions

Table 1. Description of the notations we used in the Sec. 3 (main paper) to describe our proposed SpecGuard.

$$S_{HH}(j,m,n) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I(u,v)\cdot\Big[2^{j/2}\psi^D(2^j u - m) \cdot\psi^D(2^j v - n)\Big]\, du\, dv, \quad \text{//Substitution}$$

$$S_{HH}(j,m,n) = \sum_{p=0}^{l-1}\sum_{q=0}^{l-1} T_{m,n}\cdot\psi^D(2^j u - m) \cdot\psi^D(2^j v - n), \quad \text{//Discretization}$$

$$W_\psi^d(j,u,v) = \frac{1}{l}\sum_{m=0}^{l-1}\sum_{n=0}^{l-1} T_{m,n}\cdot\psi^D\left(m - u\cdot 2^{-j}, n - v\cdot 2^{-j}\right), \quad \text{//Normalized}$$

3.2. Maximum Theoretical Watermark Capacity

To determine the maximum theoretical watermark capacity of SpecGuard, we analyze the SpecGuard's embedding pipeline, which integrates wavelet projection and spectral projection. The capacity derivation considers three key stages: 'wavelet projection,' 'spectral projection,' and 'watermark distribution,' with each stage affecting the number of available coefficients for embedding.

Impact of Wavelet Projection. SpecGuard applies wavelet projection at decomposition level L, dividing the image into sub-bands. The watermark is embedded in the high-

Activation Function	Radius (r)	PSNR↑	SSIM↑	BRA↑
ReLU	r(50)	39.54	0.93	0.97
	r(75)	38.64	0.91	0.93
	r(100)	37.96	0.91	0.95
Tanh	r(50)	37.18	0.89	0.82
	r(75)	35.33	0.85	0.78
	r(100)	37.66	0.90	0.80
LeakyReLU	r(50)	39.77	0.96	0.98
	r(75)	40.28	0.97	0.98
	r(100)	**42.89**	**0.99**	**0.99**

Table 2. Performance evaluation of SpecGuard for different radius size and activation functions while the Strenth Factor is 20.

Activation Function	Strength Factor (s)	PSNR↑	SSIM↑	BRA↑
LeakyReLU	s(5)	40.79	0.98	0.97
LeakyReLU	s(10)	39.51	0.96	0.97
LeakyReLU	s(15)	38.14	0.95	0.99
LeakyReLU	**s(20)**	**42.89**	**0.99**	**0.99**

Table 3. Impact of Strength Factor for the best combination of the activation function (LeakyReLU) and radius $r(100)$.

frequency sub-band, which retains fine image details and ensures robustness against low-frequency distortions. The spatial dimensions of the wavelet sub-band are reduced by a factor of 2^L along both height and width, resulting in a down-sampling effect.

The number of available coefficients after wavelet decomposition is as follows:

$$N_{\text{WP}} = \frac{H \times W}{4^L}, \tag{7}$$

where H and W are the image height and width, respectively. Including all image channels c, the total number of wavelet coefficients available for embedding is as follows:

$$N_{\text{WP,total}} = \frac{H \times W \times c}{4^L}. \tag{8}$$

Thus, increasing the decomposition level L reduces the available spatial coefficients exponentially, limiting embedding capacity.

Impact of Spectral Project. SpecGuard employs spectral projection using FFT to distribute the watermark in the spectral domain. The spectral coefficients are selectively utilized based on an adaptive mask that prioritizes mid-to-high-frequency components while avoiding low frequencies (which contain most perceptual information) and extremely high frequencies (which are prone to compression loss).

The fraction of spectral coefficients selected for watermarking is denoted as f_{spectral} where spectral coefficients are used in between 20% and 50% as follows:

$$0.2 \leq f_{\text{spectral}} \leq 0.5. \tag{9}$$

After spectral projection following Eq. (8), the number of coefficients available for embedding is as follows:

$$N_{\text{SP}} = f_{\text{spectral}} \times N_{\text{WP,total}} = f_{\text{spectral}} \times \frac{H \times W \times c}{4^L}. \tag{10}$$

A higher f_{spectral} increases embedding capacity but may reduce robustness to compression and noise, while a lower f_{spectral} focuses on the most resilient coefficients but limits capacity.

Watermark Distribution and Final Capacity. The watermark is distributed across the selected spectral coefficients f_{spectral} using a weighting scheme, where each coefficient can embed multiple bits. Let N_b represent the number of watermark bits per selected coefficient f_{spectral}. The total embedded bits are then as follows:

$$C_{\text{total}} = N_b \times N_{\text{SP}}. \tag{11}$$

Substituting N_{SP}, the final maximum theoretical watermark capacity of SpecGuard is as follows:

$$C_{\max}(H, W, c, L, f_{\text{spectral}}, N_b) = \frac{H \times W \times c}{4^L} \times f_{\text{spectral}} \times N_b. \tag{12}$$

The watermark capacity scales proportionally with the image dimensions $H \times W$ and the number of channels c, ensuring that larger images provide greater embedding space. However, higher wavelet decomposition levels L reduce the available capacity exponentially due to the 4^L down-sampling effect. The fraction of spectral coefficients selected for embedding, denoted as f_{spectral}, controls how much of the frequency domain is utilized, balancing capacity and robustness. Additionally, the bit depth N_b determines the number of bits embedded per coefficient, directly influencing the total watermark payload.

Thus, SpecGuard achieves a flexible balance between capacity and robustness by leveraging adaptive spectral selection and wavelet decomposition, ensuring resilience under various transformations and attacks.

4. Impact of Hyperparamters

The performance of SpecGuard is influenced by several key hyperparameters, including the activation function, radius size (r), and strength factor (s). Each parameter plays a vital role in balancing the trade-off between perceptual quality, robustness, and watermark recovery accuracy. In addition to the ablation studies shown in Section 4.5 in the main paper, here we analyze the effect of the hyperparameters individually by conducting experiments under controlled conditions and report the findings in Tab. 2 and Tab. 3. All the experiments presented here were conducted using a 128-bit watermark message.

4.1. Activation Function and Radius

Table 2 highlights the performance of SpecGuard with various activation functions, including ReLU [2], Tanh [8], and LeakyReLU [12], while keeping the strength factor s fixed at 20. Among these, LeakyReLU outperforms others in terms of PSNR, SSIM, and bit recovery accuracy values across different radius sizes. Notably, with a radius r of 100, LeakyReLU achieves a PSNR and SSIM of 42.89 and 0.99, respectively, with a bit recovery accuracy of 0.99. Overall, the results indicate the effectiveness of LeakyReLU for robust and invisible watermarking compared to ReLU and Tanh. While testing with different r, such as 50 and 75, we

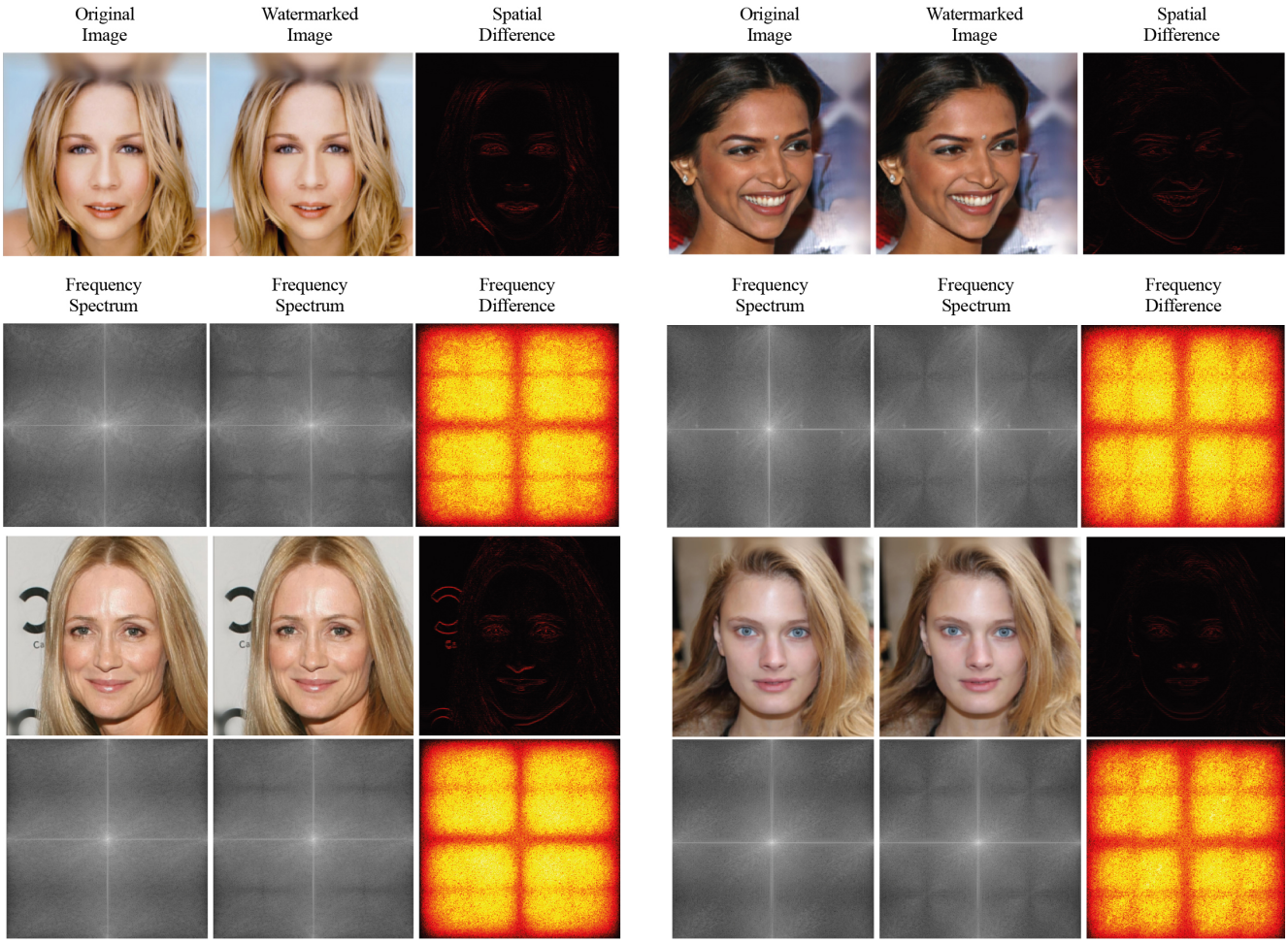

Figure 2. Visualization of the watermarking process using SpecGuard. The first row shows the original image, the watermarked image, and their spatial difference. The spatial difference highlights the minimal perceptual change between the original and watermarked images, ensuring imperceptibility. The second row presents the frequency spectrum of the original and watermarked images, along with their frequency difference, emphasizing the subtle embedding of the watermark in the high-frequency components. The comparison confirms that SpecGuard achieves invisible watermarking while maintaining robust frequency-domain characteristics for effective bit recovery.

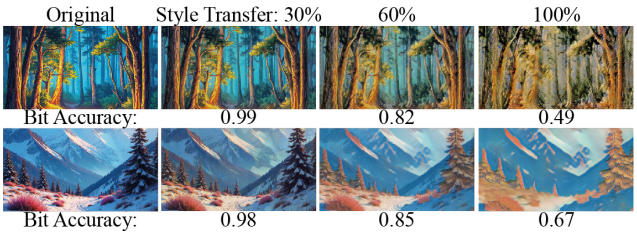

Figure 3. Effect of style transfer severity on bit recovery accuracy. As style intensity increases, bit accuracy decreases, showing the impact of major transformations.

observed a slightly lower perceptual quality and bit recovery accuracy. Therefore, we propose the SpecGuard with a combination of LeakyReLU, r of 100 and s of 20.

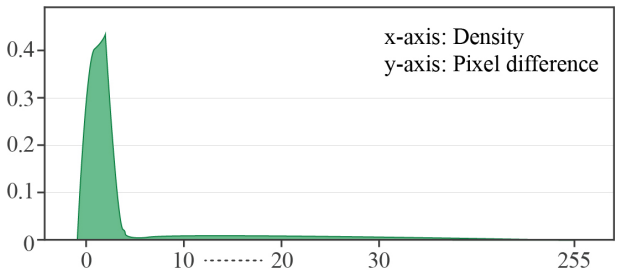

Figure 4. Pixel difference distribution between the original and watermarked images. The x-axis represents the pixel intensity difference, and the y-axis indicates the density. Most pixel differences remain close to zero, highlighting SpecGuard's minimal perceptual loss and superior imperceptibility of the embedded watermark.

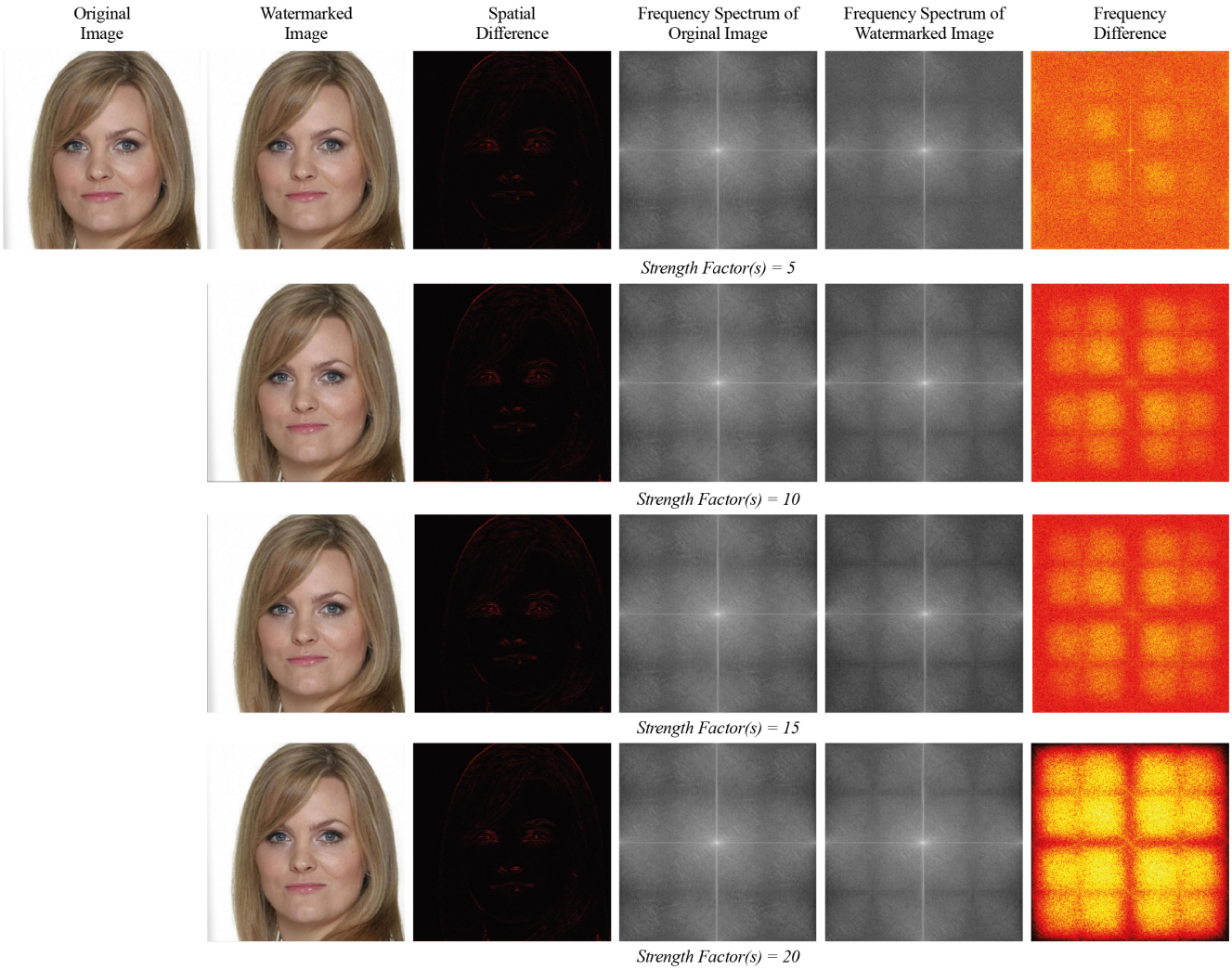

Figure 5. Visualization of the watermarking process using SpecGuard for different strength factors (s). The first row illustrates the original image, the watermarked image, and their spatial difference for $s = 5$, followed by the frequency spectra of the original and watermarked images and their frequency difference. The subsequent rows demonstrate the impact of increasing the strength factor ($s = 10, 15, 20$) on the frequency difference, highlighting the progressive embedding intensity. Higher strength factors increase the visibility in the frequency domain while maintaining imperceptibility in the spatial domain, ensuring robust watermarking without compromising image quality.

4.2. Strength Factor

Table 3 investigates the impact of the strength factor (s) using the best combination of LeakyReLU and radius $r(100)$. A strength factor of $s(20)$ achieves optimal performance with a PSNR/SSIM of 42.89/0.99 and a BRA of 0.99. Increasing s beyond 20 reduces PSNR and SSIM values, indicating diminished perceptual quality, while lower strength factors compromise robustness. Therefore, $s(20)$ effectively balances robustness and visual quality as also shown in Fig. 2.

Figure 5 further demonstrates the effect of different strength factors ($s = 5, 10, 15, 20$) on the watermark embedding process. The first row showcases the original image, the watermarked image, and their spatial difference, highlighting the imperceptibility of the watermark in the spatial domain. The subsequent rows compare the frequency spectrum of the original and watermarked images, as well as the frequency difference, illustrating how increased strength factors enhance the visibility of the watermark in the frequency domain while maintaining imperceptibility in the spatial domain. Illustrate the robustness and adaptability of the proposed SpecGuard model in embedding and retaining watermark information under varying conditions.

Attack Name	Description	Parameters
	Distortion Attacks	
Rotation	Rotates an image by a specified angle to test watermark robustness against geometric transformations.	Angle: 9° to 45° clockwise
Crop	Crops a portion of the image and resizes it back, simulating common editing.	Crop Ratio: 10% to 50%
Bright	Adjusts image brightness to test watermark stability under illumination changes.	Brightness Increase: 20% to 100%
Contrast	Modifies image contrast to simulate lighting variations.	Contrast Increase: 20% to 100%
Blur	Applies a low-pass filter to smooth the image, reducing high-frequency details.	Kernel Size: 4 to 20 pixels
Noise	Introduces random pixel fluctuations to simulate compression noise and low-quality rendering.	Std. Deviation: 0.02 to 0.1
JPEG	Compresses the image using JPEG encoding, reducing quality and adding artifacts.	Quality Score: 90 to 10
Geo	Combination of geometric distortion attacks, including rotation, crop, applied uniformly to assess cumulative effects.	Strength: Geo(x): Rotation: $9° + x \times (45° - 9°)$, Crop: $10\% + x \times (50\% - 10\%)$
Deg	Combination of degradation attacks, integrating blur, noise, and JPEG to simulate complex real-world distortions.	Strength: Deg(x): Blur: $4 + x \times (20 - 4)$, Noise: $0.02 + x \times (0.1 - 0.02)$, JPEG: $90 - x \times (90 - 10)$
	Regeneration Attacks	
Regen-Diff	Passes an image through a diffusion model to reconstruct a similar but altered version.	Denoising Steps: 40 to 200
Regen-DiffP	A prompted version of diffusion-based regeneration, leveraging text guidance to refine results.	Denoising Steps: 40 to 200 with Prompt
Regen-VAE	Uses a variational autoencoder to encode and decode an image, affecting watermark integrity.	Quality Level: 1 to 7
Regen-KLVAE	Uses a KL-regularized autoencoder to compress and reconstruct an image, weakening watermark signals.	Bottleneck Sizes: 4, 8, 16, 32
Rinse-2xDiff	Applies a two-stage diffusion regeneration, progressively altering the image over multiple steps.	Timesteps: 20 to 100 per diffusion
Rinse-4xDiff	Performs four cycles of diffusion-based image reconstruction, aggressively erasing watermark traces.	Timesteps: 10 to 50 per diffusion
	Adversarial Attacks	
AdvEmbG-KLVAE8	Embeds adversarial perturbations using a grey-box VAE-based attack to reduce detection accuracy.	KL-VAE Encoding, ϵ = 2/255 to 8/255, PGD Iterations = 100, Step Size = $0.01 \times \epsilon$
AdvEmbB-RN18	Uses a pre-trained ResNet18 model to introduce adversarial noise and affect watermark recognition.	ℓ_∞ Perturbation: 2/255 to 8/255, PGD Iterations = 50, Step Size = $0.01 \times \epsilon$
AdvEmbB-CLIP	Attacks the CLIP image encoder to introduce embedding shifts that disrupt watermark decoding.	ℓ_2 Perturbation Norm = 2.5, PGD Iterations = 50, Learning Rate = 0.001
AdvEmbB-KLVAE16	Uses an alternative KL-VAE model to introduce structured perturbations into the embedding process.	KL-VAE Embedding, Latent Size = 16, ℓ_∞ Perturbation = 4/255
AdvEmbB-SdxlVAE	Attacks Stable Diffusion XL's VAE encoder to alter latent representations and remove watermarks.	Targeted VAE Perturbation, Diffusion Steps = 100, ℓ_2 Perturbation = 3.0
AdvCls-UnWM&WM	Trains a surrogate detector on watermarked and non-watermarked images to bypass watermark detection.	Dataset Size = 3000 Images (1500 Per Class), ResNet-18, Learning Rate = 0.001, Batch Size = 128
AdvCls-Real&WM	Trains an adversarial classifier using real and watermarked images to classify watermark presence.	Dataset Size = 15,000 Images (7500 Per Class), Adam Optimizer, Learning Rate = 0.0005, Batch Size = 128, Epochs = 10
AdvCls-WM1&WM2	Exploits watermark signal variations between different users to remove or alter hidden information.	Two Sets of Watermarked Images, Model = Vision Transformer (ViT), PGD Attack, Perturbation Strength = 6/255

Table 4. Overview of attack types, their mechanisms, and key parameters based on the prior study [1] that we also utilized in our study.

Table 5. Robustness comparison of SpecGuard component configurations under four common perturbations: horizontal/vertical flip, downscaling (0.75×), and saturation increase (+40%). We report PSNR and Bit Recovery Accuracy (BRA) under each condition. The full configuration (WP + SP + adaptive θ) consistently achieves the highest robustness and fidelity across all settings, demonstrating the complementary benefits of spectral-domain embedding and adaptive thresholding.

Config	No Attack		Flip (avg. H/V)		Scale 0.75×		Satur +40%	
	PSNR↑	BRA↑	PSNR↑	BRA↑	PSNR↑	BRA↑	PSNR↑	BRA↑
WP (fixed θ)	35.3±0.4	0.92±0.01	9.2±0.5	0.25±0.05	31.2±0.3	0.65±0.03	29.3±0.4	0.63±0.03
SP (fixed θ)	36.6±0.4	0.93±0.01	11.5±0.6	0.33±0.05	32.6±0.3	0.70±0.03	30.8±0.4	0.68±0.03
WP + SP (fixed θ)	38.8±0.3	0.93±0.01	13.9±0.5	0.48±0.04	34.2±0.3	0.85±0.02	32.8±0.3	0.83±0.02
WP + SP + θ (Full)	**42.9±0.2**	**0.99±0.005**	**16.2±0.4**	**0.60±0.04**	**35.3±0.3**	**0.94±0.02**	**34.6±0.3**	**0.92±0.02**

Mean ± standard deviation over three random seeds per configuration. ***WP:*** *Wavelet Projection,* ***SP:*** *Spectral Projection.*

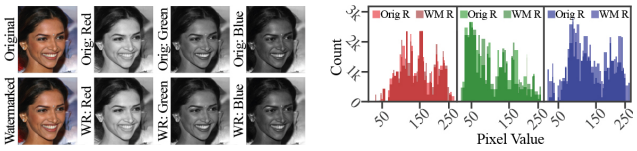

Figure 6. $\Delta R/G/B$ maps of original vs. watermarked images.

5. Visual Analysis of Watermarked Images

To further support the claim of imperceptibility, we provide a visual and channel-wise analysis of the original and watermarked images in Fig. 6. The left panel shows side-by-side comparisons of the original and watermarked versions, along with their individual R, G, and B channels. The differences are visually negligible, indicating minimal perceptual impact from the embedding process.

The right panel presents histograms of pixel intensities for each color channel before and after watermarking. The distributions of red, green, and blue intensities remain highly consistent between the original and watermarked images. These results validate that SpecGuard preserves low-level color statistics and visual fidelity across all channels, aligning with the high PSNR and SSIM values reported in the main paper.

6. Additional Ablation Studies

To further understand the contribution of each component of SpecGuard, we conducted ablation experiments across different architectural configurations and evaluated their robustness under a range of perturbations, including horizontal/vertical flips, downscaling, and saturation increase. The results are summarized in Tab. 5.

Applying only Wavelet Projection (WP) or Spectral Projection (SP) with a fixed threshold provides moderate robustness under distortions such as flip (BRA = 0.25–0.33) and scaling (BRA = 0.65–0.70). Combining WP and SP without a learnable threshold further improves recovery, particularly under geometric distortions (e.g., Flip BRA = 0.48, Scale BRA = 0.85).

The full configuration of SpecGuard, which includes WP, SP, and a learnable threshold guided by Parseval's theorem, achieved the highest robustness across all categories. For instance, under flip perturbations, the BRA improved from 0.48 to 0.60. Similarly, under saturation enhancement, the BRA improved from 0.83 to 0.92. Notably, this improvement was achieved while maintaining high fidelity under no attack (PSNR = 42.9 ± 0.2, BRA = 0.99 ± 0.005).

These results confirm the complementary roles of wavelet-domain localization and spectral-domain embedding, with the adaptive threshold enabling reliable bit recovery under challenging distortions. Overall, the full SpecGuard architecture balances imperceptibility and robustness more effectively than any other partial configuration.

7. Description of Benchmarking Attacks

To comprehensively evaluate watermark robustness, we benchmark performance against a diverse set of attacks, including distortions, regeneration, and adversarial manipulations. These attacks, derived from prior benchmarking efforts [1], assess the stability of watermarks under real-world transformations. The results are presented in Tab. 3 (main paper) and the details of the attacks are in Tab. 4, comparing multiple state-of-the-art (SOTA) methods such as Tree-Ring [11], Stable Signature [5], and StegaStamp [10]. The attacks are categorized as follows:

7.1. Distortion Attacks

These include standard image-processing transformations that alter the spatial or color properties of images. We consider rotation (9° to 45°) where images are rotated at varying degrees to test watermark stability. Resized cropping (10% to 50%) removes portions of an image and resizes the remaining content, mimicking common real-world editing. Random erasing (5% to 25%) replaces regions with gray pixels, simulating object removal. Brightness adjustments (20% to 100%) and contrast modifications (20% to 100%) simulate lighting variations. Gaussian blur (4 to 20 pixels)

applies low-pass filtering, while Gaussian noise (0.02 to 0.1 standard deviation) adds random pixel fluctuations, simulating compression noise [1].

7.2. Regeneration Attacks

These attacks leverage generative models such as diffusion and variational autoencoders (VAEs) to reconstruct images while suppressing embedded watermarks. We evaluate single regeneration attacks including Regen-Diff (diffusion-based reconstruction), Regen-DiffP (perceptually optimized diffusion), Regen-VAE (autoencoder-based reconstruction), and Regen-KLVAE (KL-regularized VAE reconstruction). Additionally, multi-step regeneration attacks such as Rinse-2xDiff and Rinse-4xDiff involve iterative diffusion processes designed to further erase watermark traces [9, 13].

7.3. Adversarial Attacks

These attacks attempt to deceive watermark detectors through embedding perturbations or surrogate model training. Grey-box embedding attacks (AdvEmbG-KLVAE8) perturb watermarks while preserving image content. Black-box embedding attacks (AdvEmbB-RN18, AdvEmbB-CLIP, AdvEmbB-KLVAE16, AdvEmbB-SdxlVAE) introduce noise during watermark embedding to decrease detection confidence. Adversarial classifiers (AdvCls-UnWM&WM, AdvCls-Real&WM, AdvCls-WM1&WM2) use learned classifiers to distinguish watermarked images and remove hidden signals [3, 6, 7, 9].

Overall, our evaluation framework ensures a rigorous assessment of watermark robustness under various real-world transformations and adversarial strategies.

Acknowledgments

This work was partly supported by Institute for Information & communication Technology Planning & evaluation (IITP) grants funded by the Korean government MSIT: (RS-2022-II221199, RS-2022-II220688, RS-2019-II190421, RS-2023-00230337, RS-2024-00356293, RS-2024-00437849, RS-2021-II212068, RS-2025-02304983, and RS-2025-02263841).

References

[1] Mucong Ding, Tahseen Rabbani, Bang An, Aakriti Agrawal, Yuancheng Xu, Chenghao Deng, Sicheng Zhu, Abdirisak Mohamed, Yuxin Wen, Tom Goldstein, et al. Waves: Benchmarking the robustness of image watermarks. In *ICLR 2024 Workshop on Reliable and Responsible Foundation Models*, 2024. 6, 7, 8

[2] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323. JMLR Workshop and Conference Proceedings, 2011. 3

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 8

[4] SS Kelkar, LL Grigsby, and J Langsner. An extension of parseval's theorem and its use in calculating transient energy in the frequency domain. *IEEE Transactions on Industrial Electronics*, (1):42–45, 1983. 1

[5] Fernandez Pierre, Guillaume Couairon, Hervé Jégou, Matthijs Douze, and Teddy Furon. The stable signature: Rooting watermarks in latent diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22466–22477, 2023. 7

[6] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 8

[7] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 8

[8] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation, parallel distributed processing, explorations in the microstructure of cognition, ed. de rumelhart and j. mcclelland. vol. 1. 1986. *Biometrika*, 71(599-607):6, 1986. 3

[9] Mehrdad Saberi, Vinu Sankar Sadasivan, Keivan Rezaei, Aounon Kumar, Atoosa Chegini, Wenxiao Wang, and Soheil Feizi. Robustness of ai-image detectors: Fundamental limits and practical attacks. *arXiv preprint arXiv:2310.00076*, 2023. 8

[10] Matthew Tancik, Ben Mildenhall, and Ren Ng. Stegastamp: Invisible hyperlinks in physical photographs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2117–2126, 2020. 7

[11] Yuxin Wen, John Kirchenbauer, Jonas Geiping, and Tom Goldstein. Tree-rings watermarks: Invisible fingerprints for diffusion images. In *Advances in Neural Information Processing Systems*, pages 58047–58063. Curran Associates, Inc., 2023. 7

[12] Bing Xu. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*, 2015. 3

[13] 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. *Advances in Neural Information Processing Systems*, 37:8643–8672, 2025. 8