A Study on the Comparative Analysis of Embedded and Zero Watermarking for Unstructured Image Protection

Jung-Min Park[1], Si-Young Nam[1], Jae-Yeong Woo[1]
and Hye-Young Kim[2,*]

[1]*Department of Games, Graduate School, Hongik University, Korea*
[2]*Department of Game Software, School of Games, Hongik University, Korea*
E-mail: kalsbold@mail.hongik.ac.kr; timezero6545@mail.hongik.ac.kr; dnwodud2574@mail.hongik.ac.kr; hykim@hongik.ac.kr
[*]*Corresponding Author*

Received 19 June 2025; Accepted 12 August 2025

Abstract

Embedded watermarking and zero-watermarking, both of which are technologies for copyright protection of digital images, are being actively studied based on their respective advantages and disadvantages. As the prevalence of copyright infringement and the use of irregular-resolution images in game content and media applications increases, there is a growing need for practical and robust image protection techniques. In this paper, we implement discrete wavelet transform (DWT)-discrete cosine transform (DCT)-singular value decomposition (SVD)-based embedded watermarking and hash-based XOR-based zero-watermarking algorithms in Python for about 200 non-standard images with various resolutions and shapes (resolution range: 512 $\times$ 512 to 6800 $\times$ 4000) and quantitatively compare and analyse their performance. We evaluate the robustness and restoration rate of each watermarking method by applying various attacks such as JPEG compression, blur, Gaussian noise, cropping, and rotation based on peak signal-to-noise ratio (PSNR), structural similarity index measure (SSIM), and normalized correlation (NC)

Journal of Web Engineering, Vol. 24_7, 1133–1154.
doi: 10.13052/jwe1540-9589.2475

indices. The experimental results show that the embedded method showed fast processing speed and stable quality maintenance performance even in high-resolution images, and zero watermarking had the advantage of not damaging the original image, but was relatively prone to being affected by restoration sensitivity and execution time. While not directly implemented in this study, the findings provide a foundational reference for integrating robust watermarking mechanisms into blockchain and InterPlanetary File System (IPFS) based copyright authentication systems, particularly for protecting high-resolution or irregularly shaped visual assets.

Keywords: Watermark, embedded watermark, zero watermark, copyright, image processing.

1 Introduction

The rapid spread of the Internet and digital content has dramatically increased the utilization of visual assets such as images. However, at the same time, copyright infringement, such as unauthorized copying and illegal distribution of images, has become a serious issue. Image copyright protection technology has been developed to respond to these issues, and among them, digital watermarking technology is one of the most widely studied protection techniques [1]. Digital watermarking generally inserts copyright information (watermark) into the original image to claim ownership. In this case, embedded watermarking inserts information by modifying the actual image data, so it is important to design it so that the damage to visual quality is minimized [2]. On the other hand, zero watermarking does not change the original image at all, but extracts image features and combines them with external watermark information to generate copyright information [3]. This method has an advantage in terms of preserving the original image, but is generally more sensitive and computationally expensive [4]. Most of the existing studies focus on standard-sized, stereoscopic images, and in practical applications, high-resolution or various non-standard resolution images are often used. In particular, images with non-standard resolutions and complex visual components are frequently used in game content, media design, and high-dimensional photographic data. In such an environment, identifying the actual performance differences of watermarking techniques is a key factor in determining the practicality of copyright protection systems. This paper aims to compare and analyse the performance of embedded watermarking and zero watermarking for non-standard images of various sizes and shapes. A total

of 200 experimental images were collected, and the resolution ranged from 512 × 512 to a maximum of 6800 × 4000. Each watermarking algorithm was implemented in Python, and after applying various image attacks such as JPEG compression, blur, noise, cropping, and rotation, robustness, restoration rate, and quality loss were evaluated based on indicators such as Peak Signal-to-noise ratio (PSNR), structural similarity index measure (SSIM), and normalized correlation (NC). This study does not simply compare the performance differences of watermarking algorithms, but also analyzes how factors such as image complexity, logo information amount, and resolution affect each method. In addition, the experimental results will contribute to providing basic judgment criteria for linking embedded or zero watermarking technology to future blockchain and InterPlanetary File System (IPFS) based copyright certification systems [5].

This study offers the following key contributions:

1. Unlike prior studies that primarily focused on standard resolution images or limited datasets, this work evaluates two watermarking techniques using approximately 200 non-standard, high-resolution images, thereby assessing their real-world applicability and limitations in diverse content environments.
2. The study implements both discrete wavelet transform (DWT)-discrete cosine transform (DCT)-singular value decomposition (SVD-based embedded watermarking and hash-XOR-based zero watermarking in Python, and provides a direct experimental comparison using the same image dataset and eleven types of image attacks under three intensity phases, enabling fair and reproducible benchmarking.
3. The robustness of both techniques is visualized and analyzed using tabulated NC values and comparative bar charts, offering insights into each method's strengths – embedded watermarking for structural consistency, and zero watermarking for precise ownership verification under clean conditions.

2 Related Work

2.1 Embedded Watermark (Based on DWT-DCT-SVD)

Embedded watermarking is a method of directly embedding a mark in a specific region (usually a display or block-based region) of the original image. In this area, a combination of DWT, DCT and SVD has been studied together as a way to achieve robust performance [1, 2].

- DWT enhances resistance to compression attacks because it allows multiple representations of an image.
- DCT changes energy intensity, so the embedded information is relatively robust to adjustment.
- SVD is characterized by the main left–right independent regression power (inverse transformation) of the image.

This embedding approach focuses on baseband components and is designed with consideration for the human visual system (HVS) to maintain imperceptibility.

Variants of DWT-DCT-SVD hybrid models have also been proposed to improve robustness and imperceptibility, notably by Roy [6] and Lai and Tsai [7], who demonstrated improved resilience under compression and noise-based attacks. Several medical watermarking studies have adopted similar DWT-DCT-SVD frameworks [10].

2.2 Zero Watermark (Hash-based Feature Extraction)

Zero watermarking is a method that does not modify the original image and obtains ownership through externally generated authentication information by combining the image's feature information and watermark. There are methods based on SVD-based feature extraction of original images or Wavelet + Hash + XOR structure [3, 4].

- Generally, the image is extracted in the form of a feature (hash) or binary vector.
- A watermark is generated through a unique logo image and XOR mosaic, etc.
- The feature extraction is repeated using the same predefined procedure, after which the watermark is reconstructed and compared with the original for authentication.

Although this method is heavier than the insertion method, it is gaining attention in the fields of medicine, litigation, and high-precision copyright because it allows ownership claims without damaging the original.

Intelligent edge-based detection techniques have also been incorporated into watermarking for smart digital environments [14].

Recently, deep learning-based approaches such as ZWNet [11], ResNet-integrated zero watermarking [12], and context-encoder-based frameworks [17] have shown promise in further improving robustness and semantic feature fidelity.

VGG19-based deep learning models have shown promising results in robust zero watermarking [16].

Foundational works in GAN training and image generation have influenced feature extraction research [18].

2.3 Types of Attacks on Image Watermarking

The performance of a watermarking algorithm depends not only on simple insertion and restoration performance, but also on robustness against various external modifications (attacks).

The representative attack types are listed in Table 1. For each type, the attack type, description, impact on the image, and impact on watermarking are listed.

To secure resistance to such attacks, many algorithms utilize DWT, NSCT, Zernike moment, feature mapping, block normalization, etc., and feature restoration techniques based on deep neural networks are also being actively studied [8, 9].

Recent research has increasingly focused on enhancing robustness against copy and tampering attacks [23, 24]. AI-generated watermarks have also been shown to be vulnerable to relatively simple attacks [24], raising concerns about their reliability in adversarial environments. In response, new benchmark datasets such as WAVES have been introduced to rigorously evaluate watermark robustness under various threat models [25]. These developments highlight the need for adaptive and hybrid watermarking strategies that combine spatial and frequency domain techniques to improve resistance against tampering and replication [26].

3 Algorithm Design and Implementation

The algorithms and test codes were implemented in Python, and about 200 images were used. The resolution of the images ranged from 512×512 to 6800×4000. The inserted watermark used a 64×64 image. The attack on the images was carried out in three stages.

3.1 Embedded Watermark Algorithm

In the proposed watermarking technique, a DWT, DCT, and SVD based hybrid watermarking technique is formulated. In this subsection, we describe the watermark embedding and extraction process by using a flowchart and algorithmically [2].

Table 1 Type of image attacks

Attack Type	Description	Effect on Image	Effect on Watermarking
Gaussian noise	Adds normal distribution noise with mean 0 and variance σ^2 to pixels	Random noise occurs around pixels, overall 'grainy' distortion	Little effect on low-frequency area-based watermarking (DWT, etc.), fatal to high-frequency watermarking
JPEG compression	Removes high-frequency components using lossy compression	Blurring of boundaries/text, block artifacts occur	Great effect on frequency-based embedded watermarking (DCT series)
Rotation	Rotates the image by a specific angle	Visually maintains the shape, but moves the entire pixel location	Deadly to feature extraction (zero watermarking) based on spatial coordinates, NC decreases sharply without normalization
Gaussian blur	Smoothing by averaging the pixel surroundings	Borders are blurred, clarity is reduced	High-frequency embedded watermarks are weakened and zero watermarking also deteriorates NC due to loss of detailed features
Cropping	Cutting out a portion of the image	Loss of some of the overall information (structural damage)	Embedded locations may disappear, so embedded types are vulnerable, but zero watermarking can be restored when normalized
Resizing	Reducing or enlarging the image	Loss of detailed information or interpolation is generated by adjusting the number of pixels	Both are distorted, but zero watermarking can be supported by normalization
Brightness	Adding/reducing a certain amount to all pixel values	Overall, becoming brighter or darker	No direct structural damage, but feature intensity can be weakened
Contrast	Expanding or reducing the difference in brightness between pixels	Bright parts are brighter, and dark parts are changed to darker	Affects the sensitivity of boundary values of feature extraction, does not significantly affect embedded type
Salt and pepper noise	Sets random pixels to 0 or 255	White/black dots appear all over the screen	Randomly generated extreme pixels interfere with feature value calculation
Sharpen	Emphasis on boundary, emphasizes pixel differences compared to surroundings	Improves clarity, but boundary distortion may occur	Affects boundary-based feature extraction (zero), high-frequency areas may distort embedded watermark
Gamma	Nonlinearly converts pixel values	Dark areas become darker, bright areas become brighter $\rightarrow$ overall contrast distortion	Affects global-based feature extraction because the overall intensity distribution changes

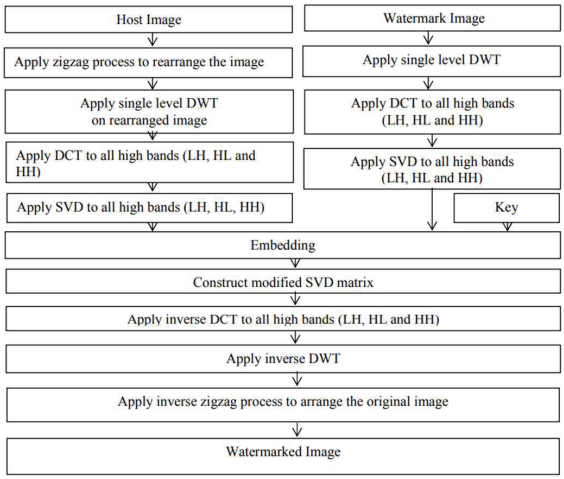

Figure 1 Watermark embedding process [2].

Algorithm 1 Watermark embedding using DWT-DCT-SVD [2]

Input : Host image HI, watermark image WI, embedding strength α
Output : Watermarked image WI
Steps:
1: Rearrange HI using zigzag scan $\rightarrow$ get RI
2: Apply 1-level DWT on RI $\rightarrow$ get LL, HL, LH, HH
3: Select high-frequency bands LH, HL, HH
4: Apply DCT to LH, HL, HH
5: Apply SVD to each DCT band $\rightarrow$ get SH1, SH2, SH3
6: Apply 1-level DWT to WI $\rightarrow$ get LL1, HL1, LH1, HH1
7: Select high bands LH1, HL1, HH1 from WI
8: Apply DCT to LH1, HL1, HH1
9: Apply SVD to each DCT band of WI $\rightarrow$ get SW1, SW2, SW3
10: Modify singular values using:
$\quad$ SHi$'$ = SHi + $\alpha \times$ SWi, for i = 1 to 3
11: Reconstruct each high-frequency band using modified SVD:
$\quad$ LH$'$, HL$'$, HH$'$ $\leftarrow$ inverse SVD(DCT_LH, DCT_HL, DCT_HH)
12: Apply inverse DCT to LH$'$, HL$'$, HH$'$
13: Apply inverse DWT using LL and modified LH$'$, HL$'$, HH$'$
14: Apply inverse zigzag to restore original spatial layout
15: Return WI as the watermarked image

Figure 1 and Algorithm 1 are watermark embedding techniques based on DWT-DCT-SVD. They sequentially apply DWT, DCT, and SVD to the original and watermarked images, respectively. The embedding is then performed by multiplying the singular values of the watermark by the embedding

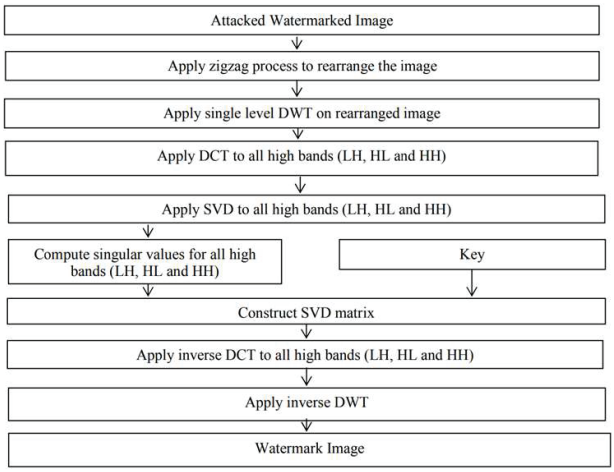

Figure 2 Watermark extraction process [2].

Algorithm 2 Watermark extraction using DWT-DCT-SVD [2]

Input : Watermarked image WI, original singular values SH1, SH2, SH3 (used during embedding), embedding strength α
Output : Extracted watermark image WI'
Steps:
1: Rearrange WI using zigzag scan $\rightarrow$ get RI*
2: Apply 1-level DWT on RI* $\rightarrow$ get LL*, HL*, LH*, HH*
3: Select high-frequency bands LH*, HL*, HH*
4: Apply DCT to LH*, HL*, HH*
5: Apply SVD to each DCT band $\rightarrow$ get SH1*, SH2*, SH3*
6: Extract watermark singular values using:
$\text{SWi}' = (\text{SHi}^* - \text{SHi}) / \alpha$, for i = 1 to 3
7: Reconstruct DCT bands for watermark using SWi'
8: Apply inverse DCT to reconstructed LH1*, HL1*, HH1*
9: Apply inverse DWT to reconstruct watermark image wi'
10: Return WI' as extracted watermark

coefficient (α) and adding them. The embedding is performed in the high-frequency band to minimize visual artifacts, and the SVD-based architecture ensures high robustness against compression and noise.

Figure 2 and Algorithm 2 represent the procedure for restoring the embedded watermark. The same transformations (DWT, DCT, and SVD) are performed on the embedded image to extract singular values. Then, based on the difference between the original singular values used during embedding, the watermark singular values are inverted. This reconstructs the

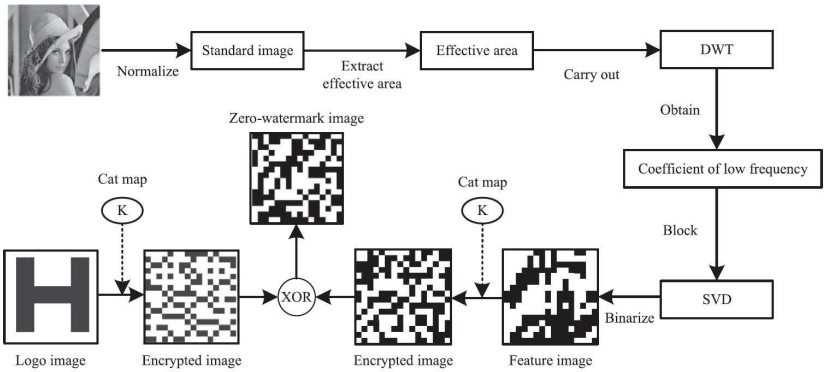

Figure 3 Zero-watermark generation process [3].

Algorithm 3 Zero-watermark generation [3]

Input : Image I, binary watermark logo W, secret chaotic key K (for scrambling)
Output: Zero-watermark V, scrambled watermark W′, scrambled feature map F′
Steps:
1: Normalize I to fixed dimensions (e.g., 512 × 512)
2: Extract the central N × N region based on image centroid
3: Apply multi-level DWT to the region → extract LL sub-band
4: Divide LL into n × n blocks
5: For each block: a. Apply SVD → extract largest singular value s_i
6: Construct binary feature map F: For each s_i: − F[i] = 1 if MSB(s_i) is odd; else 0
7: Scramble F using chaotic map with key K → get F′
8: Scramble logo W using same map → get W′
9: Compute zero-watermark V = F′ XOR W′
10: Store V, K, and reference logo for verification

high-frequency components of the watermark, and the final watermark image can be restored through inverse transformation. The accuracy of the restored watermark is evaluated through NC (normalized correlation) with the original watermark.

3.2 Zero Watermark Algorithm

In this study, the zero-watermarking algorithm was designed to ensure robustness while preserving the original image. However, to enhance reproducibility and clarify the robustness analysis, it is necessary to describe the specific computational steps in more detail.

The watermark generation phase begins by normalizing the input image to a fixed scale to ensure invariance under geometric distortions such as rotation

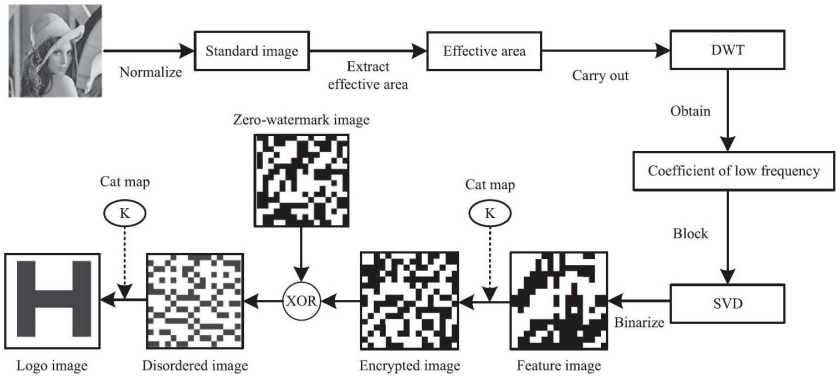

Figure 4 Watermark verification process [3].

Algorithm 4 Zero-watermark verification [3]

Input: Query image I_q, stored zero-watermark V, secret key K, reference logo W
Output: NC value indicating verification success
Steps:
1: Repeat feature extraction on I_q $\rightarrow$ get F_q
2: Scramble F_q using key K $\rightarrow$ get F_q$'$
3: Compute W_q$'$ = F_q$'$ XOR V
4: Apply inverse chaotic scrambling on W_q$'$ using K $\rightarrow$ get W_q
5: Calculate NC between W_q and reference logo W
6: If NC $\geq$ threshold (e.g., 0.9), consider watermark verified

and scaling [3]. The effective region is then extracted using the centroid of the normalized image, typically forming a square area of N $\times$ N. A multi-level discrete wavelet transform (DWT) is applied to this region to isolate the low-frequency band, which concentrates most of the image's energy and remains stable under common attacks [13].

The resulting low-frequency subband is divided into n $\times$ n sub-blocks. For each block, singular value decomposition (SVD) is performed to extract the largest singular value. These values are compiled into a matrix, and a binary feature image is generated by applying a rule such as parity check on the highest bit of each singular value (i.e., if odd $\rightarrow$ 1, else $\rightarrow$ 0) [19].

This binary feature image is then scrambled using a chaotic map such as a Cat map, with the scrambling parameters forming a secret key for authentication. A binary watermark logo (e.g., 64 $\times$ 64) is also scrambled using the same key. The final zero-watermark is generated by performing an XOR operation between the scrambled feature image and the scrambled logo image [3].

In the verification phase, the same feature extraction process is applied to the query image and the resulting scrambled feature image is XORed with the stored zero-watermark to regenerate the scrambled logo. By reversing the chaotic map using the saved key, the original logo is restored. The similarity between the restored logo and the reference logo is calculated using normalized correlation (NC). If the NC value exceeds a predefined threshold (e.g., 0.9), the watermark is considered successfully authenticated [4, 13].

This step-by-step specification provides a complete and reproducible basis for analyzing the zero-watermarking algorithm and comparing it to embedded methods.

4 Experiments and Analysis

4.1 Experimental Environment and Dataset

The experiments in this study were performed based on Python 3.10 in a Windows 11 64-bit environment, and the main libraries used were OpenCV, NumPy, PyWavelets, SciPy, and the time module. The number of test images used in the experiment was more than 170, and the resolution included various sizes from 512×512 to a maximum of 6800×4000. The inserted watermark logo was a binary logo with a size of 64×64 and used a form that mixed signature and text-based elements.

The PSNR (peak signal-to-noise ratio) is used to quantitatively assess the image quality after watermark insertion in the insertion-based watermarking technique. The SSIM (structural similarity index) measures the structural similarity between the original and watermarked images, thereby evaluating perceptual visual quality. NC (normalized correlation) evaluates the restoration accuracy by calculating the correlation coefficient between the original and the extracted watermark logos.

It is important to note that, in the case of zero-watermarking, PSNR and SSIM are not applicable since the original image is not altered. Therefore, the analysis for zero-watermarking focuses on NC values and execution time.

4.2 Attack Scenario

The robustness of the watermarking algorithm was evaluated by individually applying the transformation attack in Table 2 to each image:

To reduce visual redundancy, all 11 attack types are grouped into three consolidated figures based on attack severity (levels 1 to 3). Figures 5, 6, and 7 contain representative results for all attacks occurring at the same parameter

Table 2 Scenario of image attacks

Attack Type	Parameter
Gaussian Noise	var = 10, 20, 30
JPEG Compression	Quality = 30,40,50
Rotation	15°, 30°, 45°
Gaussian Blur	ksize = 5, 7, 9
Cropping	40%, 50%, 60% area
Resizing	20%, 30%, 40%
Brightness	Delta = 20, 30, 40
Contrast	Factor = 1.25, 1.5, 1.75
Salt & Papper Noise	Amount = 0.01, 0.05, 0.1
Sharpen	Weight = 1, 1.1, 1.2
Gamma	V = 0.2, 0.3, 0.4

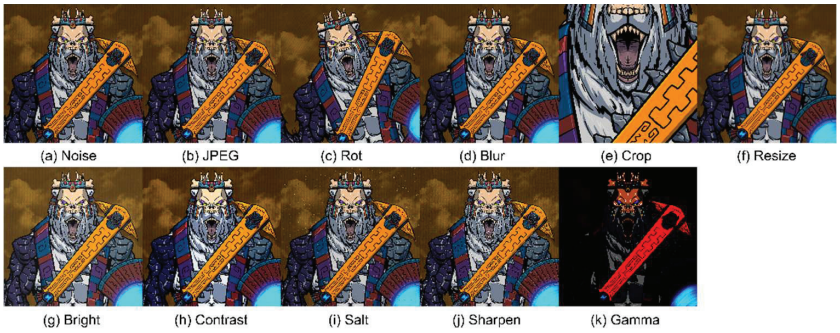

Figure 5 Phase 1 attack sample.

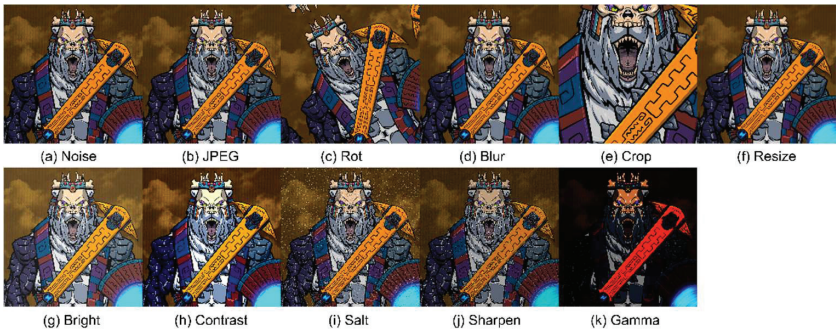

Figure 6 Phase 2 attack sample.

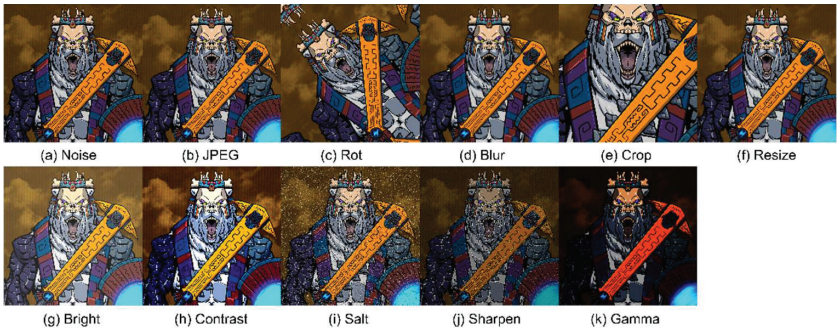

Figure 7 Phase 3 attack sample.

level. This allows readers to quickly compare the overall impact of various transformations on watermark visibility and image quality.

This consolidated layout helps summarize the effect of increasing attack severity on image appearance and sets a consistent visual basis for interpreting restoration performance metrics in Section 5.

To evaluate the resilience of watermarking schemes under immersive or complex media transformations, recent studies have proposed extended robustness indicators including geometric distortions and projection-based changes [27].

In some recent studies, NSCT-SVD [15] and Zernike-DCT [20] based techniques have also been applied to improve resilience against geometric distortions and maintain accuracy under medical or high-resolution imagery.

Noise and copy attacks remain primary evaluation factors in watermarking robustness tests [21, 22].

4.3 Experiment Result

The experiments described in Section 4.2 were designed to assess the robustness of two watermarking approaches: embedded watermarking and zero watermarking. The results are presented in Figures 8 and 9, and Tables 3 and 4.

According to Figure 8 and Table 3, the embedded watermarking technique exhibits relatively low overall restoration accuracy. Nonetheless, the normalized correlation (NC) values across different types of attacks remain consistently stable, indicating reliable structural robustness. Despite limited fidelity, this consistency suggests that the embedded watermark can still be

Table 3 Average for each phase of the attack (embedded)

Attack	Phase 1	Phase 2	Phase 3
Noise	0.619	0.6213	0.6177
Jpeg	0.5662	0.571	0.5745
Rot	0.5829	0.5878	0.5939
Blur	0.5443	0.4945	0.4812
Crop	0.5717	0.51	0.5707
Resize	0.514	0.5061	0.5422
Bright	0.5891	0.5897	0.5891
Contrast	0.5688	0.5688	0.5666
Salt	0.6214	0.6224	0.6224
Sharpen	0.57	0.57	0.57
Gamma	0.5451	0.5424	0.5444

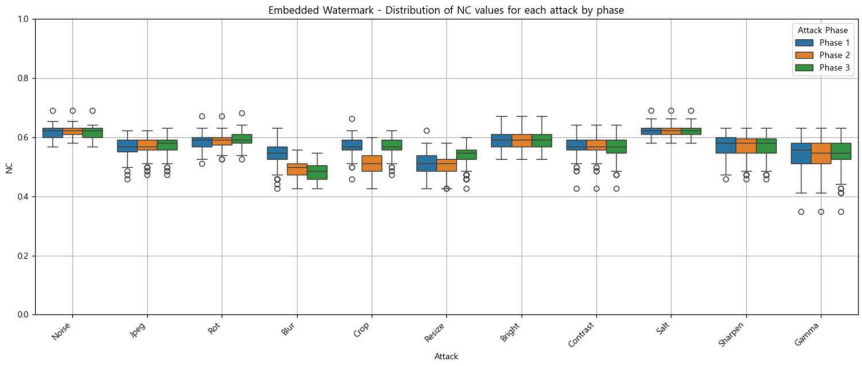

Figure 8 Result of embedded watermark attack test.

extracted even under tampering or distortion, making it suitable for applications such as copyright infringement tracking or similarity-based content identification. These findings align with prior research – particularly the hybrid DWT-DCT-SVD scheme by Agarwal and Chandel [28] – which also demonstrated strong resilience to compression and noise attacks.

In contrast, zero watermarking achieves significantly higher restoration accuracy overall, primarily because the original image remains unaltered during the watermark generation process. However, it demonstrates considerable vulnerability to geometric transformations – such as rotation and cropping – and to global intensity shifts like gamma correction. As shown in Table 4 and Figure 9, NC values under these attack types exhibit wide variation, reflecting diminished robustness. Nevertheless, the consistently high NC values under

Table 4 Average for each phase of the attack (zero)

Attack	Phase 1	Phase 2	Phase 3
Noise	0.9958	0.9958	0.9959
Jpeg	0.9976	0.9981	0.9982
Rot	0.729	0.6779	0.6503
Blur	0.9995	0.9991	0.9988
Crop	0.6423	0.6523	0.6703
Resize	0.9967	0.9977	0.998
Bright	0.9522	0.9522	0.9099
Contrast	0.9576	0.9316	0.9152
Salt	0.9952	0.9837	0.9686
Sharpen	0.9939	0.9878	0.9785
Gamma	0.7077	0.7692	0.8177

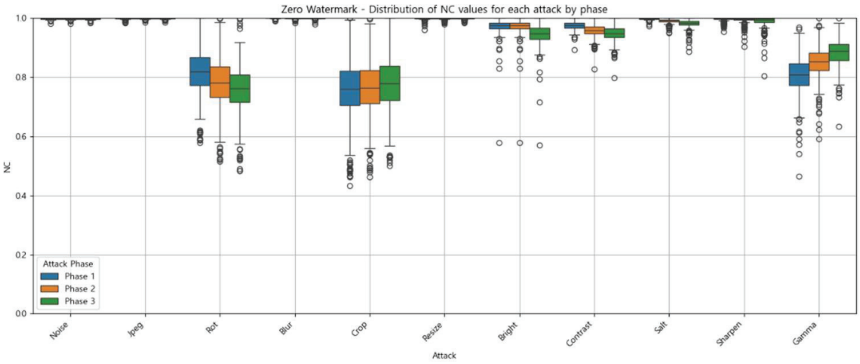

Figure 9 Result of zero watermark attack test.

non-geometric attacks (often exceeding 0.95) make this method particularly effective for applications requiring exact ownership verification or duplicate detection.

Figures 8 and 9 provide a comparative visualization of robustness across eleven types of image attacks at three intensity levels (Phases 1–3). Embedded watermarking yields NC values ranging approximately from 0.48 to 0.62, maintaining stable performance across all attack types – including geometric and compression distortions – which underscores its dependable structural integrity. Conversely, zero watermarking maintains near-perfect NC scores under non-geometric attacks such as noise, compression, and blur, but experiences sharp performance degradation under geometric distortions and gamma correction.

Although Tables 3 and 4 present the NC values separately for each method, direct comparison is valid because identical test parameters and datasets were applied. Similarly, while Figures 8 and 9 are presented independently, they reflect results from the same experimental conditions, allowing for consistent cross-referencing. This comparative analysis highlights the complementary strengths of each approach: embedded watermarking provides stable, moderate robustness under a broad range of attacks, while zero watermarking delivers high restoration accuracy in benign environments but is less resilient under structural distortions.

While the study briefly mentions game-related content as an example of high-resolution and irregularly shaped real-world images, its primary focus is on general-purpose evaluation of watermarking techniques under realistic digital content conditions. The experimental setup – utilizing more than 200 images of varying resolutions and systematically applied image distortions – was designed to simulate practical usage scenarios across diverse visual structures. Accordingly, this study contributes by establishing a robust experimental baseline for evaluating the comparative strengths and limitations of embedded and zero watermarking in visually complex and structurally varied environments.

5 Conclusion

This study conducted an experimental comparison of embedded watermarking (DWT-DCT-SVD-based) and zero watermarking (Hash + XOR-based) techniques using images with diverse and irregular resolutions. The robustness of each method was quantitatively evaluated under eleven types of image attacks.

Embedded watermarking demonstrated strong resistance to typical lossy attacks such as JPEG compression and Gaussian noise, while maintaining visual quality. It also exhibited low computational overhead and fast execution, making it suitable for real-time or large-scale applications. Although it showed vulnerability to geometric distortions like rotation and cropping, its NC values remained consistently stable across all attack types. This highlights its applicability in tasks such as copyright tracking and unauthorized usage detection, even under partial data loss.

Zero watermarking, by preserving the original image, offers advantages in legal credibility and data integrity. Feature extraction based on normalization provided robustness to certain geometric transformations. However, its performance declined sharply under specific attacks such as rotation, cropping,

and gamma correction. Despite this, its high NC values in clean conditions make it well-suited for applications requiring precise authentication, such as counterfeit detection and duplicate registration control.

In conclusion, this study highlights the complementary nature of embedded and zero watermarking techniques, suggesting that a hybrid application strategy can offer enhanced protection for unstructured image data. In practical scenarios such as NFT platforms, zero watermarking can be effectively applied for ownership registration and verification, whereas embedded watermarking serves as a reliable tool for tracing unauthorized redistribution and misuse post-deployment.

To further advance the applicability of these techniques in real-world digital rights management (DRM) systems, we intend to pursue the development of robust watermarking algorithms capable of withstanding both benign manipulations and adversarial attacks. Our future work will focus on addressing critical challenges such as structural distortion and unauthorized replication, with the goal of contributing to the design of more secure and resilient DRM frameworks.

Acknowledgement

This research was supported by Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism in 2024(Project Name: Global Talent Training Program for Copyright Management Technology in Game Contents, Project Number: RS-2024-00396709, Contribution Rate: 100%).

References

[1] Varghese, J., Bin Hussain, O., Subash, S., and Abdul Razak, T. (2023). An effective digital image watermarking scheme incorporating DCT, DFT and SVD transformations. PeerJ Computer Science.

[2] Rahman, M. M. (2013). A DWT, DCT and SVD Based Watermarking Technique to Protect the Image Piracy. International Journal of Managing Public Sector Information and Communication Technologies, 4(2), 21–32.

[3] Liu, Y., Li, Y., Wang, Z., and Li, S. (2021). *Image copyright protection based on blockchain and zero-watermark*. Journal of Intelligent & Fuzzy Systems, 41(1), 2021–2032.

[4] Zhang, Y., Liu, S., and Wang, Q. (2020). *A zero-watermark algorithm for copyright protection of remote sensing image based on blockchain*. Computers, Materials & Continua, 65(3), 2345–2359.
[5] Khan, M. I., Rahman, M. M., and Sarker, M. I. H. (2013). Digital Watermarking for Image Authentication Based on Combined DCT, DWT and SVD Transformation. International Journal of Computer Applications, 72(20), 1–6.
[6] Roy, R. (2012). Robust Image Watermarking based on DCT-DWT-SVD Method. International Journal of Computer Applications, 58(21), 1–6.
[7] Lai, C.-C., and Tsai, C.-C. (2010). Digital image watermarking using discrete wavelet transform and singular value decomposition. IEEE Transactions on Instrumentation and Measurement, 59(11), 3060–3063.
[8] Bhatnagar, G., and Raman, B. (2009). A new robust reference watermarking scheme based on DWT-SVD. Computer Standards & Interfaces, 31(5), 1002–1013.
[9] Thakkar, F. N., and Srivastava, V. K. (2017). A blind medical image watermarking: DWT-SVD based robust and secure approach for telemedicine applications. Multimedia Tools and Applications, 76(3), 3669–3697.
[10] Priyanka, Kumar, P., and Tewari, R. G. (2017). Security of Medical Images by Watermarking using DWT-DCT-SVD. International Journal of Engineering Research & Technology (IJERT), 6(4), 467–470.
[11] Liu, Y., Li, H., and Zhang, Y. (2024). ZWNet: A Deep-Learning-Powered Zero-Watermarking Scheme with Robustness and Security. Applied Sciences, 14(1), 435.
[12] Xu, H., et al. (2023). Zero-Watermark Scheme for Medical Image Protection Based on Style Feature and ResNet. Biomedical Signal Processing and Control, 86, 104810.
[13] Wang, R., et al. (2020). A Novel Zero-Watermarking Scheme Based on Variable Parameter Chaotic Mapping in NSPD-DCT Domain. IEEE Access, 8, 182391–182411.
[14] Xu, H. (2021). Digital Media Zero Watermark Copyright Protection Algorithm Based on Embedded Intelligent Edge Computing Detection. Mathematical Biosciences and Engineering, 18(5), 6771–6789.
[15] Amiri, R., and Mirzakuchaki, S. (2022). A novel zero-watermarking scheme based on NSCT-SVD and blockchain for video copyright. EURASIP Journal on Wireless Communications and Networking, 2022(1), 20.

[16] Han, B., et al. (2021). Zero-watermarking algorithm for medical image based on VGG19 deep convolution neural network. Journal of Healthcare Engineering, 2021, 1–10.
[17] Arévalo-Ancona, R. E., et al. (2024). Secure Medical Image Authentication Using Zero-Watermarking Based on Deep Learning Context Encoder. Computación y Sistemas, 28(1), 1–10.
[18] Ganguly, K. (2017). Learning Generative Adversarial Networks: Next Generation Deep Learning Simplified. Packt Publishing.
[19] Dong, P., et al. (2005). Digital watermarking robust to geometric distortions. IEEE Transactions on Image Processing, 14(12), 2140–2150.
[20] Yang, C., et al. (2021). Robust zero watermarking algorithm for medical images based on Zernike-DCT. Security and Communication Networks, 2021, 1–10.
[21] Wang, S., et al. (2019). Robust blind watermarking against Gaussian noise. Multimedia Tools and Applications, 78(18), 25745–25766.
[22] Barr, J., et al. (2003). Using Digital Watermarks with Image Signatures to Mitigate the Threat of the Copy Attack. Proceedings of ICASSP, 2003, 1–4.
[23] Kutter, M., et al. (2000). The Watermark Copy Attack. Proceedings of the SPIE, 3971, 371–380.
[24] Feizi, S., et al. (2023). Researchers Tested AI Watermarks – and Broke All of Them. Wired.
[25] WAVES: Benchmarking the Robustness of Image Watermarks. (2024). arXiv preprint arXiv:2401.08573.
[26] Deguillaume, F., et al. (2003). Secure hybrid robust watermarking resistant against tampering and copy attack. Signal Processing, 83(10), 2133–2170.
[27] Kim, Y., Kim, S.-Y., Kamyod, C., and Park, B. (2023). Proposition of robustness indicators for immersive content filtering. Journal of Web Engineering, 22(4), 731–756.
[28] Agarwal, H., and Chandel, G. S. (2022). Design of a Hybrid Digital Watermarking Algorithm with High Robustness. Journal of Web Engineering, 21(7), 2243–2261.

Biographies

Jung-Min Park is a Ph.D. candidate in the Department of Game at Hongik University, South Korea. He received his master's degree from the Department of Game at Hongik University in 2019, and his research interests include game content copyright protection, game servers, and distributed servers.

Si-Young Nam is a master's degree candidate in the Department of Game at Hongik University, South Korea. He graduated from the Department of Game Software at Hongik University in 2024 and his research interest is copyright protection of game content.

Jae-Yeong Woo is a master's degree candidate in the Department of Game at Hongik University, South Korea. He graduated from the Department of Game Software at Hongik University in 2024 and his research interest is copyright protection of game content.

Hey-Young Kim received her Ph.D. degree in Computer Science and Engineering from the Korea University, South Korea in February 2005. During her Ph.D. studies, she focused on location management schemes and traffic modeling for mobile IPv6, cellular network and network mobility. She developed a network protocol for 9 years while working as a senior researcher at Hyundai Electronics. She has been working as a Full Professor at Hongik University, South Korea since March 2007. Her research interests include traffic modeling, load balancing schemes and copyright technology for digital content on blockchain and web3.