



Deep Learning and Error Correction Code-Based Robust Watermark Extraction from DWT–Chaotic Embedding

Jantana Panyavaraporn¹ and Paramate Horkaew^{2,*}

¹Department of Electrical Engineering, Faculty of Engineering, Burapha University, Chonburi, 20231, Thailand.

²School of Computer Engineering, Suranaree University of Technology, Nakhon Ratchasima, Thailand.

*Corresponding author: phorkaew@sut.ac.th

Received 13 November 2025

Revised 10 May 2026

Accepted 20 May 2026

Abstract

A blind digital watermarking framework based on Discrete Wavelet Transform (DWT) is proposed for secure image authentication and copyright protection. A binary watermark is embedded within the LH subband, and a chaotic map is used to select embedding positions, ensuring high randomness and security. During the extraction, a convolutional neural network (CNN) is applied to the embedded subband and combined with an error correction code to improve recovery performance under attacks. Experimental results on 200 COCO test images demonstrate that, with a magnitude factor of 0.35, the proposed method achieves an average Peak Signal-to-Noise Ratio (PSNR) of 29.46 dB and an SSIM of 0.9189 on 200 COCO test images. The results indicate that the framework achieves stable watermark recovery under blur, Gaussian noise, JPEG compression, and scaling, with high perceptual image quality.

Keywords: Convolutional neural network, Chaotic map, Error correction code, Watermark

1. Introduction

In recent decades, digital multimedia has rapidly become an important means of knowledge exchange and technology development in various fields of engineering and applied science. Digital platforms have been used to transmit images, videos, satellite images, medical images, sensor data, etc. While these data help create innovation, they also raise concerns over copyright violation, unauthorized distribution, and improper modification. Ensuring the reliability and authenticity of digital information has become a major challenge for academia, business, and industry [1].

An effective means of tackling this challenge is digital watermarking. In contrast to traditional cryptographic methods that protect data during transmission, it embeds invisible information directly into the host media. The hidden information can later be extracted to verify ownership, identify modifications, or authenticate data [2]. Watermarking can be advantageous in most applications, where multimedia may undergo a series of processing operations prior to verification, such as compression, scaling, or format conversion.

In applied science and engineering, several watermarking methods are available for securing data exchange. For example, in mechanical and civil engineering, confidential design models and CAD drawings are protected against unauthorized use [3]. In electrical and computer engineering, watermarking secures data transmission in sensor networks and Internet of Things (IoT) applications, as information is often compressed or transmitted over unprotected channels [4]. In biomedical engineering, medical images embedded with patient-specific watermarks can maintain data reliability across hospital information systems [5].

This paper presents a robust image watermarking scheme based on DWT embedding and deep learning (DL), with integrated error correction code (ECC) in the extraction. The watermark embedding site is determined by a chaotic map to prevent informed attacks. The method is evaluated through experiments under different distortion scenarios, using standard test images.

2. Materials and Methods

2.1 Related Work

Digital watermarking is a technique for protecting multimedia content against unauthorized use, modification, and duplication. Embedding an invisible watermark into images enables the implementation of authentication, copyright protection, and verification. Traditional watermarking techniques can be classified into spatial and frequency-domain methods. The former directly manipulates pixel values, whereas the latter embeds watermarks within their transformed coefficients. The spatial-domain techniques are often simpler but less robust against compression, noise, and geometric distortion. On the contrary, frequency-domain techniques ensure higher robustness, because the watermark is embedded into significant components less susceptible to such adulterations.

Several recent studies indicate the effectiveness of using DWT as the main embedding domain for image watermarking. Lee, Seo, and Kim [6] propose a blind DWT watermarking scheme for depth-image-based rendering of free-viewpoint images, using a depth variation prediction map to select robust embedding positions. The method achieves high invisibility and robustness against compression, noise, and geometric attacks, proving its suitability for 3D image rendering. Similarly, Panyavaraporn and Horkaew [7] propose a wavelet-based watermarking method that combines DWT and Lorenz chaotic signal localization for embedding positions. The results confirm that the method is resilient against JPEG compression, filtering, zooming, and noise addition, while maintaining high visual quality. Previous studies show that DWT with chaotic-based localization strikes a balance between invisibility, robustness, and security. Similarly, Alhazmi et al. [8] propose a chaotic-based watermarking approach that embeds watermarks in the DWT domain and uses chaotic maps for localization. The scheme is designed to protect face recognition systems in public surveillance, maintaining high visual quality and enabling secure detection under various attacks.

Although watermarking schemes based only on DWT demonstrate invisibility and robustness, recent research focuses on hybrid approaches that combine DWT with other transforms to enhance performance. Begum et al. [9] present a hybrid method that combines DWT, Discrete Cosine Transform (DCT), and Singular Value Decomposition (SVD). This watermarking method is not only highly invisible but also resistant to filtering, rotation, and intensity adjustments. In addition, Tang and Zhou [10] propose a robust algorithm that uses DWT and SVD in the fractional Fourier domain, reporting high invisibility and robustness with a zero-bit error rate under Gaussian noise, JPEG compression, and cropping attacks. Ernawan, Ariatmanto, and Firdaus [11] improve DWT-DCT watermarking by adaptively modifying selected coefficients. This approach results in high visual quality and resilience against noise and filtering attacks. Cheema, Adnan, and Mehmood [12] optimize a semi-blind scheme for color images that combines finite ridgelet transform, DWT, and SVD. Particle Swarm Optimization (PSO) is used for embedding optimization, and the Arnold transform is used for enhanced security. Besides high visual quality, their approach produces watermarks that are robust against geometric and basic attacks. Hybrid approaches, combining DWT with other transforms, can resolve the trade-offs between these criteria.

The rapid advancement in DL presents a new opportunity to develop a watermarking technique that can automatically learn both embedding and extraction. Li et al. [13] present a resampling-detection-network-based approach that integrates DWT and DCT to withstand scaling and cutting attacks. The deep resampling detection network improves watermark recovery accuracy under spatial distortions. Al-Dmour et al. [14] propose a hybrid watermarking scheme using combined DWT, DCT, and PSO to optimize embedding strength. In addition, a convolutional neural network (CNN) has been used to evaluate the watermark's resistance to common attacks. Quan et al. [15] embed watermarking into a neural network model. This creates ownership signatures within the deep architectures for image processing. The scheme offers additional intellectual property protection, hence maintaining model integrity. Li et al. [16] propose a security-oriented watermarking framework for smart city applications using a Fast R-CNN model to accelerate and improve watermark detection. Zhong et al. [17] propose a fully automated scheme using deep neural networks (DNNs) with a novel loss function, enabling high payload capacity and robustness against unknown attacks, testing it with challenging camera-captured images. Padhi et al. [18] propose a dual watermarking method using DL to embed cryptographic and perceptual hashes to support copyright protection and authentication. This system maintains high image quality while resisting content-preserving manipulations. He et al. [19] combine transformer architectures with invertible neural networks and introduce an affine transformer module to enhance feature robustness. It is also resilient to compression, noise, and geometric distortion without degrading original image fidelity. The studies show DL's advancement in image watermarking for emerging digital environments.

The existing DL-based watermarking techniques remain impeded by many challenges, specifically, maintaining their robustness under distortion and ensuring correct extraction under severe signal degradation. Most current techniques rely primarily on spatial or frequency-domain feature learning but often overlook the exploitation of redundancy to enhance recovery accuracy under adverse conditions. To address these limitations, recent research focuses on incorporating error correction and adaptive feature enhancement mechanisms within

deep models to improve extraction reliability without sacrificing image quality. Therefore, this paper aims to develop a DL and ECC-based robust watermark embedding and extraction framework within the DWT domain to achieve better resilience against multi-type attacks, while maintaining visual fidelity and computational efficiency.

2.2 Proposed Framework

This section presents the overall design, experimental materials, and methodology framework of the proposed DL and ECC-based watermark extraction system. The framework was designed to enhance watermark robustness and extraction reliability via the combination of DWT, chaotic embedding, and DNNs. The section is divided into five parts: (1) the description of datasets and experimental materials, (2) the DWT–chaotic watermark embedding process, (3) the DL-based watermark extraction network, (4) the incorporation of ECC to improve decoding accuracy, and (5) the performance evaluation metrics assessing its imperceptibility, robustness, and accuracy.

2.2.1 Dataset and Experimental Materials

The experiments were performed using a subset of color images from the COCO (Common Objects in Context) dataset [20]. A total of 2,000 images were selected for training the proposed watermark extraction network, with an additional 200 separate images employed for testing. Each image was resized to 256×256 pixels for consistency with the size of the DWT subband and the CNN input structure. In addition, the Lena color image was employed as a benchmark example to quantitatively assess the embedding invisibility and extraction accuracy under various attacks. Binary and QR code images were used as the hidden embedded information. Each of these binary watermarks was resized, corresponding to the system configuration defined in the embedding process. Examples of dataset images used in the experiments are shown in Figure 1.

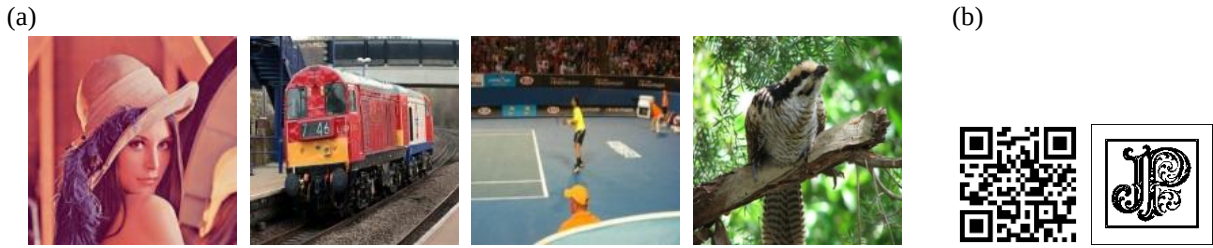


Figure 1 Example of (a) original image and (b) watermark employed in the experiments.

2.2.2 DWT–Chaotic Watermark Embedding Process

The watermark embedding process was conducted using a two-level DWT and combined chaotic position mapping. As shown in Figure 2, an original RGB image was first converted to the YCrCb color space, with the luminance component (Y component) selected for the watermark embedding plane, due to its high sensitivity and stability under compression and imaging noise. The Y component was decomposed into subbands using a two-level DWT with periodization mode to minimize border artifacts.

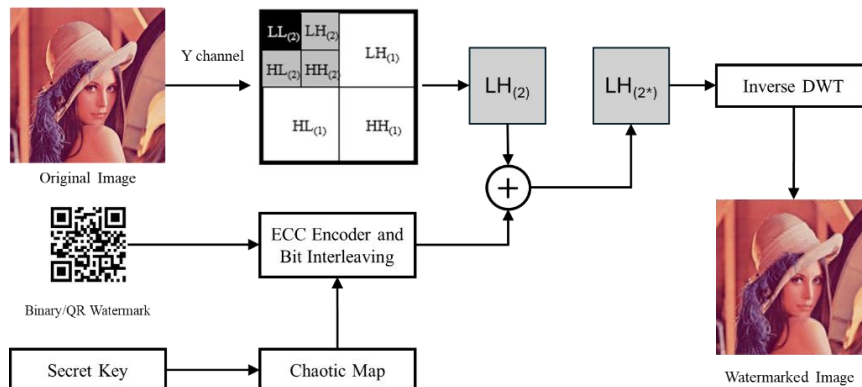


Figure 2 Block diagram of the watermark embedding process with a watermark embedded in the LH_2 subband.

From the two-level subbands, either LH_2 or HL_2 was selected for watermark embedding, providing a balance between imperceptibility and robustness. A logistic chaotic map, combined with a secret key, was used to generate pseudo-random embedding positions for watermark bits, resulting in secure and unpredictable spatial distribution.

The chaos was realized using the logistic map function, defined as $x_{n+1} = \mu x_n(1-x_n)$ with $\mu = 3.99$. The initial condition was derived from a secret key and normalized within the range $[0, 1]$. Due to high sensitivity to initial conditions of the logistic map, a small change in the key causes a different sequence of embedding positions. This characteristic enhances security by preventing unauthorized extraction without the correct key. The chaotic sequence was used to generate unique embedding positions within the LH_2 subband in the transform domain. Figure 3 shows the embedding position patterns generated in the LH_2 subband using two different mappings. The random map in (a) generates embedding locations based on a pseudo-random generator, resulting in a statistically uniform pattern. In contrast, the chaotic map in (b) is created by the logistic chaotic system, presenting an equally stable but more irregular distribution. While both maps embed the same watermark, the chaotic map produces a uniform, highly irregular spatial distribution compared to the pseudo-random map. This characteristic can improve security by reducing the predictability of embedding locations and enhance robustness during watermark extraction under attacks.



Figure 3 Embedding positions in the LH_2 subband (a) using the random map, and (b) using the chaotic map.

Each selected DWT coefficient (c_i) was modified according to the watermark bit $w_i \in \{0,1\}$ using a magnitude-based amplitude adjustment (Eq. (1)).

$$c'_i = \text{sign}(2w_i - 1) \times \max(|c_i|, \sigma \cdot \alpha) \quad (1)$$

where σ is the standard deviation of the subband coefficients, and α is the magnitude factor controlling embedding strength. This embedding strength enhances the robustness of embedded bits while maintaining image fidelity. After the watermark bits were embedded, the modified subband was reconstructed with the remaining unaltered DWT coefficients by inverse DWT (IDWT), resulting in the watermarked image. This procedure ensures that the watermark remains invisible while maintaining resistance against image-processing operations, such as JPEG compression, Gaussian noise, blurring, scaling, and rotation.

2.2.3 Deep Learning-Based Watermark Extraction

The watermark extraction process involved a CNN design to recover embedded watermark bits from a distorted or attacked watermarked image. Figure 4 shows the conversion of the image into the YCrCb color space, with only the luminance (Y) component (previously the embedding plane) being processed. Subsequently, a two-level DWT was applied using the same wavelet parameters, i.e., type and periodization mode, as prescribed during the embedding. The selected subband was then extracted as the input to the CNN-based watermark recovery.

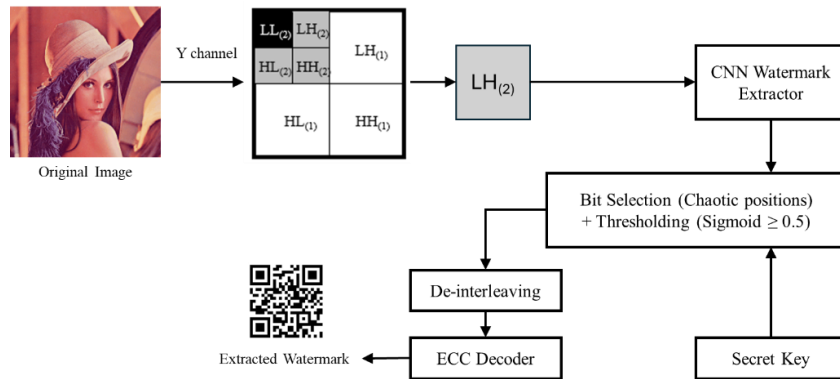


Figure 4 Block diagram of the deep learning-based watermark extraction process using the LH_2 subband as input.

The proposed model in the CNN extractor was a fully convolutional network comprising five convolutional layers with progressively increasing feature channels, and ReLU activations in each hidden layer. The network outputs a single-channel probability heatmap, with each pixel corresponding to the likelihood of a watermark bit being “1”. Each layer captures orderly frequency-domain features related to the embedded watermark pattern. Its architecture is summarized in Table 1. The CNN was designed with only five convolutional layers due to the LH₂ subband containing structured watermark features in the transformed space. Consequently, a small network is sufficient to learn robust watermark representations without incurring additional model complexity.

The input to the CNN extractor consisted of the LH₂ subband coefficients obtained from the DWT of the watermarked image. In other words, the extractor operated directly on the transformed domain, allowing the network to learn wavelet features associated with the embedded watermark. The CNN output remained in the transformed domain and represented the watermark bit probability matrices. The extracted bit sequence was then de-interleaved using the same chaotic map employed during embedding to reconstruct the correct bit positions. Finally, the reordered bit sequence was passed to the ECC decoder to correct residual errors and fully reconstruct the original watermark.

Table 1 CNN network architecture

Layer	Type	Kernel Size	Channels (in → out)	Activation
1	Conv2D	3×3	1 → 32	ReLU
2	Conv2D	3×3	32 → 64	ReLU
3	Conv2D	3×3	64 → 64	ReLU
4	Conv2D	3×3	64 → 64	ReLU
5	Conv2D	1×1	64 → 1	-

In the training phase, the CNN extractor was optimized using binary cross-entropy loss between the predicted and ground-truth bit vectors. The training dataset was dynamically augmented using a randomized attack generator to simulate common image distortions such as JPEG compression, Gaussian blur, scaling, rotation, additive noise, brightness/contrast variation, and gamma correction. This augmentation helped the network to generalize to a wide range of distortions and to learn robust watermark feature representations in the DWT.

2.2.4 Error Correction Code (ECC) Integration

To enhance the robustness of extraction against signal distortions, a Reed–Solomon (RS) ECC was utilized in the proposed framework. The RS code is a block-based linear error-correcting code; it can detect and correct symbol errors within each codeword, thereby improving the accuracy of bit recovery after attacks.

The binary watermark sequence (w) was encoded using the RS (n, k) code, where k and n denote the number of data and total symbols per block, respectively. Each group of k bytes was extended by $n-k$ parity bytes, forming a systematic codeword represented as Eq. (2).

$$C(x) = M(x) \cdot x^{n-k} + P(x) \quad (2)$$

where $M(x)$ is the message polynomial and $P(x)$ is the parity polynomial generated by the encoder. The code rate is defined as $R = k/n$, and the redundancy ($n-k$) specifies error resilience against multiple symbol errors per block.

In the experiment, each watermark bitstream was packed in multiple bytes and encoded using the RS (255, k) configuration with a varying number of parity symbols ($nsym$), i.e., 16, 24, or 32, depending on the robustness level. The RS (255, k) code was selected because it is a widely used Reed–Solomon configuration defined over $GF(2^8)$, providing symbol-level error-correction capability. The number of parity symbols was selected to balance error-correction capability and payload efficiency. Higher $nsym$ values provide greater error-correction capability and may improve robustness against distortion, but at the cost of increased redundancy. The encoded bits were interleaved using a pseudo-random sequence generated from the same secret as in the chaotic embedding process. This interleaving operation spreads nearby bits across different RS blocks, thereby reducing burst errors in the subband.

In the extraction, the CNN-based extractor developed a probability map of individual embedded bit positions. The original sequence was restored by bit de-interleaving and then decoded by RS. The correction of symbol errors was based on the computed syndrome polynomial $S(x)$ and the error locator polynomial. By setting an appropriate level of robustness, the decoder can restore the original message bits, even when multiple coefficients within the DWT domain have been corrupted or lost.

The integration of ECC improved the system’s bit error rate (BER) and normalized correlation (NC) under various attacks. As demonstrated later in the paper, the RS-coded watermark enabled reliable recovery even under JPEG compression, scaling, and noise variations, outperforming the non-ECC baseline while maintaining image

quality. Therefore, the ECC is an essential component in the DL-based extractor. It functions as the traditional signal protection layer, ensuring robust watermark reconstruction.

2.2.5 Performance Evaluations

Firstly, a similarity measure was computed between the original ($\mathbf{S}$) and extracted ($\hat{\mathbf{S}}$) images. The results reported herein are based on normalized correlation (NC), expressed in Eq. (3).

$$NC = \left(\sum_{i=1}^M \mathbf{S}_i \mathbf{S}'_i \right) / \left(\sum_{i=1}^M \mathbf{S}^2_i \right) \quad (3)$$

The quality of the watermarked image ($\tilde{I}$) compared with the original image (I) was also measured using the Peak Signal-to-Noise Ratio (PSNR), as given in Eq. (4).

$$PSNR = 10 \log_{10} \left(\left((2^b - 1)^2 \right) / \left(\frac{1}{m \times n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (I(i, j) - \tilde{I}(i, j))^2 \right) \right) \quad (4)$$

where b is the number of bits specifying pixel depth and $m \times n$ is the image dimension.

SSIM is a perceptual metric that quantifies the similarity between the original and watermarked image by incorporating luminance, contrast, and structural comparison. Moreover, the BER measures the ratio of bits incorrectly received to the total number of bits transmitted. The BER is commonly used to evaluate the embedding accuracy of communications data.

3. Results and Discussion

The proposed method was evaluated using the Lena image as the original image. A CNN-based watermark extractor was trained with 2,000 color images randomly selected from the COCO dataset, while an additional 200 distinct images were used for testing. To improve the model's robustness and generalization, the training process included data augmentation via randomized attacks. The augmentation included multiple transformations with probabilistic selection in mini-batches, including JPEG compression (quality 30–90), Gaussian blur (kernel sizes 3–7), scaling (70–100%), rotation (-15° to $+15^\circ$), and several noise types, namely, Gaussian, speckle, Poisson, and salt-and-pepper noises. In addition, brightness, contrast, and gamma variations were made to simulate signal degradations. The CNN extractor model was trained for 100 epochs with a batch size of eight and a learning rate of 0.001 using the Adam optimizer. After training, the watermark embedding and extraction process was tested on the Lena image.

During performance evaluation, image quality was measured using PSNR and SSIM to reflect the visual fidelity between the host and watermarked images. In addition, BER and NC were used to determine the watermark recovery accuracy. The statistical impact of embedding in the transform domain was analyzed by the KL divergence between LH_2 coefficient distributions of the original and watermarked images. Perceptual distortion was also quantified using Learned Perceptual Image Patch Similarity (LPIPS). Finally, the False Alarm Rate (FAR) was computed to assess the validity of the detector when processing non-watermarked images.

The watermark embedding and extraction performance on different subbands (LL_2 , LH_2 , and HL_2) was compared using a magnitude factor of 0.35. For the Lena image, the results indicated that the LH_2 subband gave the highest PSNR of 34.21 dB, SSIM of 0.9621, and successful watermark reconstruction ($NC = 1$, $BER = 0$). In contrast, the LL_2 subband showed visible artifacts and a lower PSNR (11.17 dB), suggesting greater distortion in the watermarked image due to interference with low-frequency components. Thus, the LH_2 subband was selected as the embedding domain.

Table 2 presents an ablation study comparing the proposed framework with the conventional DWT-based watermarking method in [21]. The analysis considers three configurations: embedding-site selection using random and chaotic maps, the addition of error correction, and the integration of error correction with interleaving. The comparison was performed on the Lena image and 200 COCO test images using PSNR, SSIM, KL divergence, and LPIPS. According to the results, the random-map and chaotic-map configurations achieved comparable objective image-quality metrics. For the Lena image, both approaches achieved PSNR values of approximately 34–35 dB and SSIM values above 0.96. For the 200 COCO test images, the PSNR values were approximately 29–30 dB, with SSIM values above 0.91. The KL divergence and LPIPS values also remained within a similar range across the compared configurations. These results indicate that the use of chaotic maps, error correction,

and interleaving does not substantially degrade the objective quality of the watermarked images compared with the baseline method [21]. However, the results do not suggest any consistent advantage of using chaotic maps over random maps in terms of PSNR, SSIM, KL divergence, or LPIPS under the tested conditions.

Therefore, the results in Table 2 can be empirically interpreted as the proposed components being integrable while not undermining objective image quality. The main contribution of the proposed framework lies in the extraction process, in which a CNN-based decoder is combined with coding components to support watermark recovery under attacks. This contribution statement is thoroughly validated in the subsequent robustness experiments.

Table 2 Comparison between the proposed CNN-based extraction and traditional DWT-based watermarking

Ablation study	Lena Image				200 COCO test images			
	PSNR (dB)	SSIM	KL	LPIPS	PSNR (dB)	SSIM	KL	LPIPS
1 Random Map [21]	34.46	0.9653	0.3760	0.0321	30.50	0.9289	0.3270	0.0565
Chaotic map	34.81	0.9690	0.3567	0.0307	29.86	0.9277	0.3430	0.0573
2 Random Map [21] + ECC	34.33	0.9628	0.4472	0.0355	29.92	0.9193	0.3938	0.0638
Chaotic+ ECC	34.14	0.9633	0.4325	0.0350	29.38	0.9183	0.4074	0.0640
3 Random Map [21] + ECC + Interleave	33.84	0.9621	0.4531	0.0342	29.91	0.9185	0.3934	0.0646
Chaotic+ ECC + Interleave (proposed framework)	34.21	0.9621	0.4004	0.0327	29.46	0.9189	0.4046	0.0635

*Watermark is a QR code.

Tables 3–5 present the quantitative results obtained from the Lena image using the Daubechies wavelet (db4) under different experimental conditions: subband selection, the magnitude factor, and robustness analysis against image-processing attacks. Table 3 compares the performance of the proposed framework, with chaotic map and random map embedding position strategies, and a magnitude factor of 0.35. Two watermark types, QR code and logo image, were embedded into the Lena image under the same embedding and extraction settings. The results show that chaotic and random embedding achieved similar visual quality, with the PSNR above 33 dB and SSIM above 0.96, indicating invisible watermark embedding. However, the chaotic map consistently outperformed random map embedding in terms of robustness, as reflected in the slightly higher PSNR (34.21 dB vs. 33.84 dB for QR code, 34.20 dB vs. 33.55 dB for logo image) and lower pre-ECC BER. The chaotic map generated pseudo-random but deterministic locations, uniformly distributed across the LH₂ subband, minimizing distortion in neighboring coefficients. In contrast, random embedding often clusters bits in neighboring regions, being more susceptible to distortion or compression artifacts. After ECC decoding, both methods achieved optimal recovery. This indicates the combined benefit of chaotic mapping and ECC in ensuring reliable extraction. However, chaotic-based embedding provides more consistent performance across different watermark types and thus offers superior reliability and security against potential attack prediction.

For 200 COCO test images, the mean PSNR, SSIM, NC, and BER values indicate no distinct trend from a distortion-oriented aspect. However, chaotic mapping may provide embedding-location uncertainty when the parameters are kept secret because it generates parameter-dependent embedding locations, which may be difficult to reproduce without the correct parameters. KL divergence for the chaotic method was lower in the Lena image experiments but comparable on average, suggesting comparable preservation of LH₂ coefficient statistics under this metric. Meanwhile, LPIPS remained low, indicating limited perceptual distortion under the adopted metric. In all experiments, FAR was 0, indicating that none of the 200 COCO test images were incorrectly identified as containing a watermark under the NC higher than the 0.9 decision threshold. This outcome suggests that no false correlation with the original watermark was observed for the tested non-watermarked images under this threshold.

Table 3 Comparison of chaotic and random maps for embedding positions with a magnitude factor of 0.35

Metrics	Lena Image				Metrics (Average)	200 COCO test images			
	QR code		Logo			QR code		Logo	
	Chaotic Map	Random Map	Chaotic Map	Random Map		Chaotic Map	Random Map	Chaotic Map	Random Map
PSNR (dB)	34.21	33.84	34.20	33.55	PSNR (dB)	29.4558	29.9107	29.4554	29.9847
SSIM	0.9621	0.9621	0.9632	0.9600	SSIM	0.9189	0.9185	0.9192	0.9191
NC	1	1	1	1	NC	1	1	0.9650	0.9950
BER	0	0.001	0.001	0	BER	0.0008	0.0006	0.0032	0.0025
KL	0.4004	0.4531	0.4378	0.4565	KL	0.4046	0.3934	0.4224	0.4082
LPIPS	0.0327	0.0342	0.0341	0.0433	LPIPS	0.0635	0.0646	0.0626	0.0627
FAR	0	0	0	0	FAR	0	0	0	0

Figure 5 shows the performance measured over the 200 COCO test images. As revealed by the experimental results, chaotic and random embedding achieved comparable metrics, and neither method outperformed the other across all images. The sensitivity of the chaotic map refers to key sensitivity, where a slight change in the initial condition or control parameter can generate a different sequence of embedding positions. The selected embedding locations become dependent on the secret chaotic parameters, potentially making unauthorized estimation or reproduction of the embedding pattern more difficult. Similarly, pseudo-random embedding generation can also provide embedding-location uncertainty when the random seed is kept secret. Thus, this property is not exclusively unique to chaotic maps. Since a dedicated key-sensitivity analysis has not been conducted, the security-related benefit is discussed as a theoretical property.

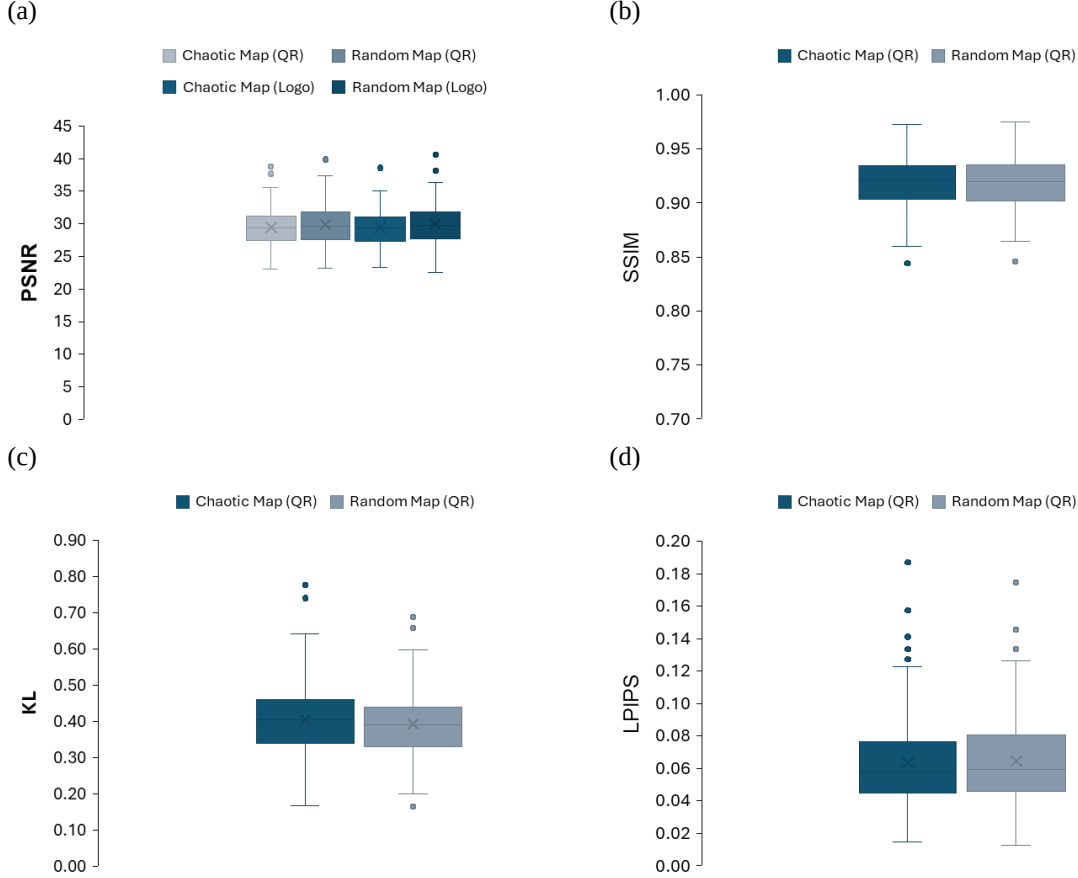


Figure 5 Comparison of (a) PSNR, (b) SSIM, (c) KL divergence, and (d) LPIPS on the 200 COCO test images.

Table 4 presents the performance of the proposed framework using the Lena image under different magnitude factors ranging from 0.30 to 0.45. The resulting PSNR and SSIM values suggest that the watermarked images maintained high objective image quality, with values above 34 dB and 0.95, respectively, across all tested settings. The NC and BER values indicate successful watermark recovery under non-attack conditions. The NC remained at 1, while BER approached zero, suggesting that the magnitude factor had a small effect on extraction accuracy within the tested range.

The proposed framework was tested against various attacks—JPEG compression, blurring, Gaussian and speckle noise, scaling, brightness adjustment, contrast modification, and gamma correction. The framework successfully extracted the embedded watermark under several tested attack conditions. The results suggest a stable extraction performance for the evaluated geometric and non-geometric distortions. In particular, the method demonstrated stability under JPEG quality factors of more than 70 and Gaussian noise with $\sigma = 2$, typically degrading performance in conventional watermarking methods.

Table 5 shows the performance of the proposed framework using different wavelet types with a magnitude factor of 0.35, using the logo image as the watermark. The tested wavelets were from the db4, sym4, bior2.2, rbio2.2, coif1, and dmey families. The sym4 and dmey gave the best visual quality, with a PSNR of 35.06 dB and 34.87 dB, respectively. Their SSIM values were above 0.96, indicating higher image quality after watermark embedding. In addition, rbio2.2 gave the lowest PSNR of 30.8 dB, suggesting a small degradation in image quality compared to other transforms. The extraction accuracy remained high across the wavelet types, with NC = 1 and

BER = 0 under the tested conditions. The low pre-ECC BER and zero post-ECC BER suggest that the error-correction component contributed to successful recovery.

In terms of robustness against attacks, db4 showed acceptable overall performance. Although sym4 and dmey provided slightly higher PSNR and SSIM values, db4 maintained a stable extraction performance under a wider range of attacks. Therefore, db4 was selected as the wavelet basis for the proposed framework. Its performance remained stable under JPEG compression with quality factors as low as 70. It also showed higher resistance to Gaussian noise ($\sigma = 2, 5$) and blur attacks (kernel sizes 3, 5, 7) compared to other bases. The system remained robust to speckle noise, brightness, contrast, and gamma variations, ensuring correct watermark recovery under intense transformations. Although other wavelets, such as sym4 and dmey, gave slightly higher PSNR and SSIM values, db4 presented the best trade-off between invisibility and robustness, resulting in it being the most suitable wavelet basis for the proposed framework.

Table 6 presents the average watermark recovery performance of the proposed framework on 200 COCO test images using the Daubechies wavelet (db4). The QR code watermark was embedded in the LH₂ subband and evaluated under different image-processing attacks. At a magnitude factor of 0.3, the average image quality was preserved, with a PSNR of 29.4936 dB and SSIM of 0.9236. Meanwhile, with a magnitude factor of 0.35, the average PSNR dropped slightly to 29.4558 dB while the SSIM was 0.9189. According to the results, increasing the magnitude factor from 0.30 to 0.35 improved recovery performance under several tested attacks, with only a small change in the average PSNR and SSIM values. However, attacks that caused severe pixel-level distortions, such as brightness adjustment and salt-and-pepper noise, remained challenging. Overall, a magnitude factor of 0.35 provided a favorable compromise between objective image quality and watermark recovery performance under the tested conditions.

Generally, digital images shared over social media platforms are automatically compressed to reduce their size and hence transmission bandwidth. This process introduces JPEG artifacts, which can degrade watermark quality. Therefore, evaluating the robustness of the proposed framework under different JPEG compression quality levels is required to mirror actual communication environments. Table 7 shows the number of successful extractions of watermarks from the 200 COCO test images under different JPEG compression levels using magnitude factors of 0.3 and 0.35. The extraction rate improves with higher JPEG quality factors. At a low quality of 30, only one of the 200 images was successfully recovered for a magnitude factor of 0.3, while six images were extracted for 0.35. As the quality factor increases, the recovery rate improves, resulting in 196 and 198 successful extractions at a quality of 90 for magnitude factors of 0.3 and 0.35, respectively. The results indicate that the proposed method can maintain higher recovery rates under medium to high JPEG quality factors.

Figure 6 presents examples of watermarked images created with a magnitude factor of 0.35. Visual inspection of the examples shows no visible watermark pattern. However, no subjective visual assessment was conducted. This figure is provided only for a qualitative illustration, while visual quality is evaluated using objective metrics such as PSNR, SSIM, KL divergence, and LPIPS. Table 8 shows examples of attacked images and corresponding extracted watermarks under different attack conditions. The extracted watermarks remained visually recognizable under several attack types, particularly Gaussian noise, JPEG compression, and scaling. However, performance degraded under rotation and Poisson noise. In addition, limited robustness was observed for salt-and-pepper noise, particularly at higher distortion levels.

The results also show the role and limitations of ECC and interleaving in the extraction process. When residual bit errors remain within the correction capability, the scheme can help recover the watermark with little or no visible noise artifacts. This differs from direct extraction without error correction, where residual bit errors may remain in the extracted watermark. However, under severe distortions, the number or distribution of bit errors may exceed its correction capability. In such failure cases, the decoding process may fail, with the extracted output potentially appearing as a black image because the watermark cannot be successfully reconstructed.

Table 4 Performance of the proposed framework under different magnitude factors

		QR Code				Logo			
		0.30	0.35	0.40	0.45	0.30	0.35	0.40	0.45
Metrics	PSNR (dB)	34.25	34.21	34.15	34.09	34.25	34.20	34.15	34.08
	SSIM	0.9639	0.9621	0.9599	0.9573	0.9651	0.9632	0.9611	0.9586
	NC	1.000	1.000	1.000	1.000	1	1	1	1
	BER (pre-ECC)	0.000	0.000	0.0005	0.0005	0.0015	0.0015	0.002	0.001
Attack Type	JPEG Quality Factors	80–100	70–100	70–100	70–100	80–100	70–100	70–100	70–100
	Blur (size)	3, 5, 7	3, 5, 7	3, 5, 7	3, 5, 7	-	3, 5, 7	3, 5, 7	3, 5, 7
	Scaling Ratio	0.7–1.0	0.7–1.0	0.7–1.0	0.7–1.0	0.7–1.0	0.7–1.0	0.7–1.0	0.7–1.0
	Gaussian Noise (σ)	2	2, 5	2, 5	2, 5	2	2, 5	2, 5	2, 5
	Speckle Noise	0.02	0.02	0.02	0.02–0.05	0.02	0.02	0.02	0.02
	Brightness	-30 to +30	-30 to +30	-30 to +30	-30 to +30	-30 to +30	-30 to +30	-30 to +30	-30 to +30
	Contrast Scale	0.8–1.2	0.8–1.2	0.8–1.2	0.8–1.2	0.8–1.2	0.8–1.2	0.8–1.2	0.8–1.2
	Gamma	0.7–1.5	0.7–1.5	0.7–1.5	0.7–1.5	0.7–1.5	0.7–1.5	0.7–1.5	0.7–1.5

Table 5 Performance of the proposed framework with a magnitude factor of 0.35 under different wavelet types

Metrics	db4	sym4	bior2.2	rbio2.2	coif1	dmey	Attack Type	db4	sym4	bior2.2	rbio2.2	coif1	dmey
PSNR (dB)	34.20	35.06	34.73	30.80	34.14	34.87	JPEG	70–100	80–100	85–100	70–100	80–100	80–100
SSIM	0.963	0.969	0.973	0.933	0.968	0.966	Blur	3, 5, 7	-	-	-	-	3, 5, 7
NC	1	1	1	1	1	1	Scaling Ratio	0.7–1.0	0.7–1.0	0.8–1.0	0.8–1.0	0.7–1.0	0.7–1.0
BER (pre-ECC)	0.0015	0.002	0.0015	0.0005	0.0005	0.0025	Gaussian Noise	2, 5	2	2	2	2, 5	2
							Speckle Noise	0.02	0.02	0.02	0.02	0.02–0.05	0.02
							Brightness	-30 to +30	-30 to +30	-30 to +30	-30 to +30	-30 to +30	-15 to +30
							Contrast Scale	0.8–1.2	0.8–1.2	0.8–1.2	0.8–1.2	0.8–1.2	0.8–1.2
							Gamma	0.7–1.5	0.7–1.5	0.7–1.5	0.7–1.5	0.7–1.5	0.7–1.5

Table 6 Performance of the proposed framework on 200 COCO test images under different attacks

Attack Type	Magnitude Factor	Average BER (post-ECC)	Average Success Rate	Attack Type	Magnitude Factor	Average BER (post-ECC)	Average Success Rate
Blur (3, 5, 7)	0.3	0.0755	0.8617	JPEG	0.3	0.0400	0.9267
	0.35	0.03	0.945	(70, 80, 90)	0.35	0.0200	0.9633
Brightness	0.3	0.2204	0.5963	Scale	0.3	0.0218	0.9600
(-30, -15, 15, 0)	0.35	0.2143	0.6075	(0.7, 0.8, 0.9)	0.35	0.0127	0.9767
Contrast	0.3	0.1237	0.7733	Speckle Noise	0.3	0.1406	0.7425
(0.8, 1, 1.2)	0.35	0.1174	0.785	(0.02, 0.05)	0.35	0.0969	0.8225
Gamma	0.3	0.0528	0.9033	Salt & Pepper Noise	0.3	0.4777	0.1250
(0.7, 1, 1.5)	0.35	0.0455	0.9167	(0.002, 0.005)	0.35	0.4463	0.1825
Gaussian Noise	0.3	0.0560	0.8975				
(2, 5)	0.35	0.0232	0.9575				

Table 7 Number of successfully extracted watermarks under different JPEG quality factors

		Magnitude Factor	JPEG Quality Factors						
			30	40	50	60	70	80	90
Number of successes	0.3		1	15	50	96	158	187	196
	0.35		6	42	96	144	185	195	198

**Figure 6** Examples of watermarked images with a magnitude factor of 0.35.**Table 8** Examples of attacked watermarked images and extracted watermarks under various attack types

Attack Type	Attack Image	Extracted Watermark	Attack Image	Extracted Watermark	Attack Type	Attack Image	Extracted Watermark	Attack Image	Extracted Watermark
Scale (0.7)					JPEG (40)				
Speckle Noise (0.02)					JPEG (70)				
Gaussian Noise (2)					Gamma (1.5)				

4. Conclusion

The proposed DWT-based watermarking framework uses LH₂ subband embedding, chaotic-map-based position selection, CNN-based extraction, and ECC with interleaving for watermark recovery. The experimental results show that the framework maintained objective image quality under non-attack conditions, with PSNR values higher than 34 dB and SSIM values greater than 0.96 for the Lena image. For the 200 COCO test images, the proposed framework achieved an average PSNR of 29.46 dB and SSIM of 0.9189.

Under the tested conditions, chaotic-map-based and random-map-based embedding achieved comparable objective image-quality metrics. In the proposed framework, the chaotic map is used as a deterministic and parameter-dependent mechanism for generating embedding positions. Its key sensitivity allows small changes in the initial condition or control parameter to produce different embedding patterns. This property supports key-dependent position selection while maintaining image quality comparable to random-map-based embedding. The proposed framework reliably recovered the watermark under several attacks, such as JPEG compression, Gaussian noise, blur, and scaling, but remained limited under severe distortions such as salt-and-pepper noise, rotation, and Poisson noise. ECC and interleaving offered error-correction capability during watermark reconstruction when residual bit errors remained within a correctable range.

Future work should further investigate the behavior of the components under more severe distortion conditions and explore adaptive magnitude factor selection to further improve robustness and image quality.

5. Acknowledgments

This work was financially supported by the Fundamental Fund (FF), National Science, Research and Innovation Fund (NSRF), fiscal year 2026, Burapha University, Thailand (Project No. 2.48/2569).

6. Author Contributions

JP: Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Software, Validation, Visualization, Writing—original draft; PH: Conceptualization, Formal analysis, Methodology, Supervision, Validation, Visualization, Writing—review and editing.

7. References

- [1] Singh S, Dhumane A. Unmasking digital deceptions: An integrative review of deepfake detection, multimedia forensics, and cybersecurity challenges. *MethodsX*. 2025;15:103632.
- [2] Naem SAS, Hameed, SM. Digital watermarking techniques, challenges, and applications: A review. *Mesop J Cybersecur*. 2025;5(2):453–476.
- [3] Zhang X, Liu S, Li X, Li J. Engineering graphics copyright protection via CAT and reversible state loop maximum length cellular automata. *J Comput Virol Hacking Tech*. 2025;21(2):12.
- [4] Zhao D, Ju L, Tong F, Yu F, Li X. Multi-level privacy preserving scheme for visual IoT data based on compressive sensing. In: *Proceedings of the 28th International Conference on Computer Supported Cooperative Work in Design*; 2025. p. 2209–2214.
- [5] Giakoumaki A, Pavlopoulos S, Koutsouris D. Secure and efficient health data management through multiple watermarking on medical images. *Med Biol Eng Comput*. 2006;44(8):619–631.
- [6] Lee Y, Seo Y, Kim D. Digital blind watermarking based on depth variation prediction map and DWT for DIBR free-viewpoint image. *Signal Process Image Commun*. 2019;70:104–113.
- [7] Panyavaraporn J, Horkaew P. Wavelet-based digital image watermarking by using Lorenz chaotic signal localization. *J Inf Process Syst*. 2019;15(1):169–180.
- [8] Alhazmi BS, Benrhouma O, AlHassan AN, Ashraf M, Alqahtany SS. A chaotic-based watermarking scheme for ensuring integrity of a face recognition system in public large gathering scenario. *J King Saud Univ Compu Inf Sci*. 2023;35(10):101845.
- [9] Begum M, Ferdush J, Uddin MS. A hybrid robust watermarking system based on discrete cosine transform, discrete wavelet transform, and singular value decomposition. *J. King Saud Univ Comput Inf Sci*. 2022;34(8):5856–5867.
- [10] Tang M, Zhou F. A robust and secure watermarking algorithm based on DWT and SVD in the fractional order Fourier transform domain. *Array*. 2022;15:100230.
- [11] Ernawan F, Ariatmanto D, Firdaus A. An improved image watermarking by modifying selected DWT-DCT coefficients. *IEEE Access*. 2021;9:45474–45485.
- [12] Cheema AM, Adnan SM, Mehmood Z. A novel optimized semi-blind scheme for color image watermarking. *IEEE Access*. 2020;8:169525–169547.

- [13] Li HL, Liu Z, Wu Q, et al. Resampling-detection-network-based robust image watermarking against scaling and cutting. *Sensors (Basel)*. 2023;23(19):8195.
- [14] Al-Dmour MM, Al-Tarawneh AH, Al-Dmour AA. Robust image watermarking using DWT, DCT, and PSO with CNN-based attack evaluation. *Int Arab J Inf Technol*. 2024;21(2):204–214.
- [15] Quan Y, Teng H, Chen Y, Ji H. Watermarking deep neural networks in image processing. *IEEE Trans Neural Netw Learn Syst*. 2021;32(5):1852–1865.
- [16] Li D, Deng L, Gupta BB, Wang H, Choi C. A novel CNN based security guaranteed image watermarking generation scenario for smart city applications. *Inf Sci*. 2019;479:432–447.
- [17] Zhong X, Huang PC, Mastorakis S, Shih FY. An automated and robust image watermarking scheme based on deep neural networks. *IEEE Trans Multimedia*. 2021;23:1951–1961.
- [18] Padhi SK, Tiwari A, Ali SS. Deep learning-based dual watermarking for image copyright protection and authentication. *IEEE Trans Artif Intell*. 2024;5(12):6134–6145.
- [19] He Z, Hu R, Wu J, Luo T, Xu H. A Transformer-based invertible neural network for robust image watermarking. *J Vis Commun Image Represent*. 2024;104:104317.
- [20] COCO dataset. Common objects in context (COCO) [Internet]. 2017. Available from: <https://cocodataset.org>
- [21] Panyavaraporn J, Horkaew P, Wongtrairat W. QR code watermarking algorithm based on wavelet transform. In: 2013 13th International Symposium on Communications and Information Technologies. IEEE; 2013. p. 791–796.