Engineering Science and Technology, an International Journal 48 (2023) 101586

Contents lists available at ScienceDirect

Engineering Science and Technology, an International Journal

journal homepage: www.elsevier.com/locate/jestch

A blind recovery technique with integer wavelet transforms in image watermarking

Ferda Ernawan [a,*], Afrig Aminuddin [b], Suraya Abu Bakar [a]

[a] Faculty of Computing, Universiti Malaysia Pahang Al-Sultan Abdullah, Pekan 26600, Pahang, Malaysia
[b] Faculty of Computer Science, Universitas Amikom Yogyakarta, Sleman 55283, Yogyakarta, Indonesia

ARTICLE INFO

Keywords:
Image watermarking
Fragile watermarking
Authentication
Blind recovery
Inpainting technique

ABSTRACT

The development of internet technology has simplified the sharing and modification of digital image information. The aim of this study is to propose a new blind recovery technique based on integer wavelets transform (BRIWT) by utilizing their image content. The LSB adjustment technique on the integer wavelet transform is used to embed recovery data into the two least significant bits (LSB) of the image content. Authentication bits are embedded into the current locations of the LSB of the image content, while the recovery information is embedded into different block locations based on the proposed block mapping. The embedded recovery data is securely placed at random locations within the two LSBs using a secret key. A three-layer embedding of authentication bits is used to validate the integrity of the image contents, achieving high precision and accuracy. Tamper localization accuracy is employed to identify recovery bits from the image content. This research also investigates the image inpainting method to enhance recovery from tampered images. The proposed image inpainting is performed by identifying non-tampered pixels in the surrounding tamper localization. The results demonstrate that the proposed scheme can produce highly watermarked images with imperceptibility, with an average SSIM value of 0.9978 and a PSNR value of 46.20 dB. The proposed scheme significantly improves the accuracy of tamper localization, with a precision of 0.9943 and an accuracy of 0.9971. The proposed recovery technique using integer wavelet transforms achieves high-quality blind recovery with an SSIM value of 0.9934 under a tampering rate of 10%. The findings of this study reveal that the proposed scheme improves the quality of blind recovery by 14.2 % under a tampering rate of 80 %.

1. Introduction

With the advancement of computer technology in editing software, users can directly modify and tamper with multimedia data. Digital watermarking techniques have become an alternative to provide copyright protection and authentication for digital images [1], audio [2] and video [3]. In the context of digital images, the information in an image can be misconstrued due to editing and modification. Skilled editing can result in an image with no discernible signs of the alteration process. Digital images are susceptible to easy alteration or tampering, which can lead to misinformation or misrepresentation [4]. This is particularly important because images are frequently used for various purposes, including as evidence in legal proceedings, for marketing, and for documentation. In response to this need, research has developed several watermarking techniques to ensure copyright protection, establish ownership, and verify the integrity of digital images. A digital image watermarking system embeds watermark information in a digital image so that the intended recipient can verify the image's originality.

Digital image watermarking can be either visible or invisible [5]. Visible watermarking is the process of directly modifying image pixels with a watermark. In the case of an invisible watermark, the watermark is inconspicuously inserted [6]. Invisible watermarking techniques have gained popularity due to their imperceptibility. Furthermore, invisible watermarking systems can be classified as (1) fragile, (2) semi-fragile, or (3) robust [7]. Fragile watermarking has a high capability for recovering data from uncompressed images. It is widely used to verify the integrity and authenticity of an image, ensuring that it has not been modified and is authentic [8]. On the other hand, semi-fragile watermark methods maintain a balance between reliability and fragility [9]. They can resist JPG compression to some extent but have limited recovery capabilities. Lastly, the robust watermarking method strikes a balance between imperceptibility and robustness [10]. This approach is commonly used

* Corresponding author.
E-mail address: ferda@umpsa.edu.my (F. Ernawan).

https://doi.org/10.1016/j.jestch.2023.101586
Received 11 February 2023; Received in revised form 1 October 2023; Accepted 20 November 2023
Available online 5 December 2023

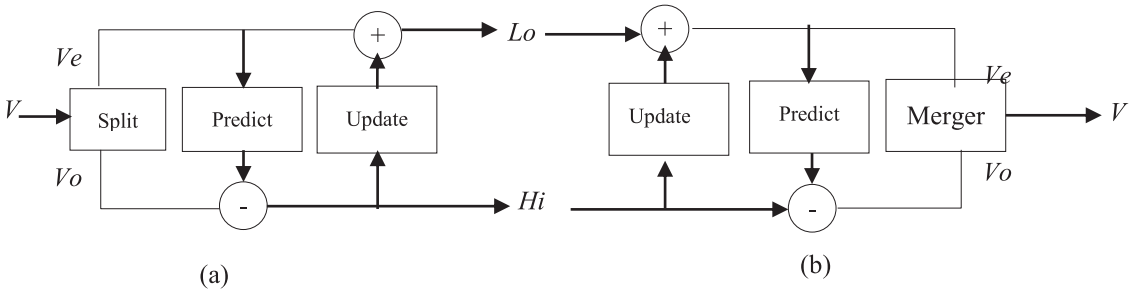

Fig. 1. (a) lifting operation block diagram, (b) inverse lifting operation block diagram.

to protect ownership and can withstand malicious modifications.

In 2020, Molina et al. [11] proposed a method for fragile authentication and recovery in image watermarking. They inserted both recovery and authentication bits into each block of the image. If any modifications were made to the block, tampering was detected. The authentication bits were inserted in LSB of the cover image, while the recovery bits were placed in different block locations based on a block mapping. If any modifications were made to the current block, the recovery bits were securely stored in a different location and could be used to recover the current block. The scheme also included an image inpainting process to enhance the quality of the recovered image. During the image inpainting process, the average block was used to replace the tampered data in the post-processing phase. The method was tested on several images and proved effective in detecting tampering and enabling the self-recovery of tampered images. The imperceptibility of the watermarked image produced a PSNR value of 45 dB. Additionally, it demonstrated robustness against various attacks, including cropping, scaling, and JPEG compression, achieving PSNR value ranging from 37.34 dB to 19.20 dB for tampering rates between 10 % and 80 %.

The recovery process can sometimes encounter a problem called tamper coincidence, which occurs when tampering also affects the location of the recovery bits. Tamper coincidence is a challenge that can arise during the recovery process in image authentication techniques [12]. It refers to a situation in which tampering, or modification of an image also impacts the location of the recovery bits, resulting in their loss. This renders the recovery of the tampered block impossible, as the necessary information is no longer available. To address this issue, Haghighi et al. [13] proposed using multiple sets of recovery data or implementing image inpainting techniques. Multiple recovery data provide additional opportunities to recover the tampered block by using different sets of recovery bits. One scheme [11] even employs three sets of recovery data to offer a third chance for recovery. Image inpainting techniques utilize information from surrounding blocks to restore the tampered block. Another approach [12] uses an image inpainting technique to recover tamper coincidence-affected blocks by utilizing recovery data from surrounding blocks.

This paper presents a method for adding a fragile watermark to images for authentication and blind recovery purposes (BRIWT). The cover image is divided into 2x2 pixels, and authentication bits of its image content are inserted into the LSB. The recovery bits are inserted into the two LSBs at random locations using a secret key. The proposed scheme uses Integer Wavelet Transform (IWT) to generate recovery data, which can replace other techniques that rely on the average block value. The recovery process involves a novel image inpainting technique that considers eight regions of the non-tampered area to replace the tampered data. The structure of this paper is as follows: The second section provides an overview of current fragile watermarking methods. The third section presents the newly proposed method for embedding and extracting a watermark for authentication and recovery. The fourth section showcases the results of tamper localization evaluation and compares the proposed technique with others, highlighting its ability to perform blind recoveries. Finally, Section 5 offers conclusions for this research.

2. Related works

2.1. Integer wavelet transform (IWT)

The IWT uses a lifting scheme to implement integer values in the block image, eliminating rounding errors in the watermark extraction scheme. The inverse of IWT can be used for the perfect reconstruction of the binary watermark image [1415]. Fig. 1 shows a representation of the lifting operation block diagram, and the steps are described as follows:

The first step in the block diagram lifting operation is the split, which divides the initial signal into even values (*Ve*) and odd values (*Vo*). The second step is the predict, in which *Vo* is helped by *Ve* and then predicted based on a predictor to obtain a new value of *Vo*. The third step is the update, in which the predicted odd values are combined with the original even samples based on an updater. The predicted odd values are considered high-frequency components, while the generated values are regarded as low-frequency components.

2.2. Existing fragile watermarking schemes

In 2013, Tong et al. [16] presented a watermarking method using a chaotic block map. The scheme employed a 2x2 pixel block map with chaotic maps to enhance security. Furthermore, the scheme improved recovery results after manipulation by combining the most significant bit (MSB) and least significant bit (LSB) to enhance tampering detection rates and defend against attacks. An optimization technique was also applied to enhance imperceptibility. The overall approach is more secure and exhibits better performance in detecting and recovering from tampering, even in cases where the tampered area is substantial. However, the recovered image still contains traces of tampered regions when the tampering rate is at 10 %.

In 2014, Dadkhah et al. [17] proposed an authentication and self-recovery method using Singular Value Decomposition (SVD). The method creates two separate tamper detection keys using SVD, with each key being unique to each image block and protected through encryption. Additionally, the method employed a block-partitioning technique for 4x4 and 2x2 blocks to enhance tamper localization performance. However, the recovered image achieved a PSNR of 22.51 dB for a 10 % tampering rate and 11.40 for a 50 % tampering rate, indicating room for improvement.

In 2016, Singh et al. [18] presented a watermarking method for authentication, which involves dividing the image into 2x2 pixels. The scheme utilized the five most significant bits (five-MSB) to generate recovery bits. The method employed a two-level tamper detection mechanism, ensuring a high probability of detecting tampered blocks. The scheme achieved a peak signal-to-noise ratio of 26.55 dB under a 10 % tampering rate. However, there is room for improvement in the quality of the recovered image.

In 2018, Fan et al. [19] introduced a self-recovery method that uses the SPIHT algorithm. Instead of applying the SPIHT algorithm to the

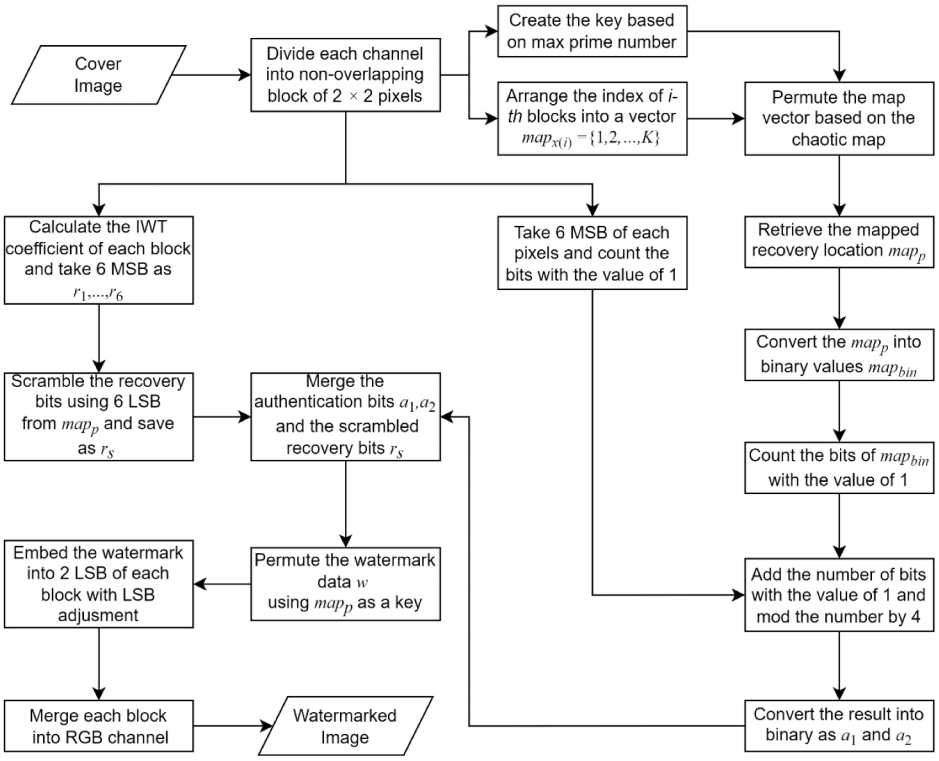

Fig. 2. The proposed BRIWT embedding watermark.

entire image, this method applies it to individual image blocks. This means that even if a portion of the encoded image bits is altered, only the corresponding image blocks will be affected, not the entire image. To safeguard the encoded image bits, the method employs repeated coding, where two versions of encoded bits for the same image block can correct each other's errors. Additionally, the scheme uses a chaotic sequence to scramble check bits, enabling it to detect tampering. However, the SPIHT transform has the drawback of producing incorrect recovery bits for a block, resulting in a lower quality recovered image for the affected block.

In 2018, Tai et al. [20] proposed a technique for adding a type of digital watermark called a fragile watermark to images. This method used IWT to generate the recovery bits, and the actual watermark data was embedded in the LSB. The recovery bits were embedded into another block according to a chaotic map. The method used wavelet transform to replace the average of the recovery data in order to minimize the smoothing and blocking effects on the recovered images. The technique for detecting tampering uses a hierarchical strategy for high accuracy and considers the 3x3 block-neighborhood. However, when the tampered area exceeds 50 %, the scheme produces low-quality recovered images.

In 2020, Sarkar et al. [21] introduced a method for detecting and restoring heavily tampered images, which is a challenging problem. They proposed two different approaches for identifying tampered regions in an image and restoring them. The first approach is based on a quadruple watermarking method, which involves dividing the image into four parts. A mapping algorithm is used to determine these regions. The second approach is based on wavelet decomposition, which involves embedding two different watermarks for tamper detection and restoration. The results showed that the scheme can achieve good restoration for tampering rates less than 50 %. However, if the tampering rate exceeds 50 %, the scheme is unable to recover the image.

In 2021, Liu and Yuan [22] proposed a method for protecting digital images from tampering and restoring the original content if tampering is detected. The authentication and recovery bits are embedded into the LSB of the image. The first authentication bit is produced using a parity bit, applied to each pixel, while the second check bit is created by applying a hashing algorithm to blocks of the image that have been divided. The combination of these two authentication bits helps minimize false-negative errors. Additionally, an additional post-processing step called Adaptive Structural Element Calculation is utilized for tamper localization. However, the process of adding the watermark and restoring the image can result in low imperceptibility of the watermarked and recovered images.

In 2022, Kosuru et al. [23] proposed a technique for detecting and correcting image tampering using a combination of Merkle trees and a method called remainder value differencing. This approach utilizes the concept of differencing and addresses issues related to boundary problems. The watermark bits are calculated from the quotients using a Merkle tree, and these bits are combined with other bits generated from a mathematical sequence to create recovery bits. These bits are then stored in the remainders by making slight modifications. When extracting the watermark, the technique can identify tampered blocks and correct them, but it is not effective for tampering above 20 %. Furthermore, there is room for improvement in the overall quality of the recovered image.

In 2022, Lin et al. [24] proposed a method for protecting and restoring digital images that combines the turtle shell algorithm and a compression technique called AMBTC. The method employs a two-layer technique to incorporate recovery bits and validation bits, which are embedded into the image using the AMBTC compression technique. The validation bits are used to identify tampered areas, while the recovery bits are utilized to restore the original image. Experimental results demonstrated that their scheme produces watermarked images with high visual quality. However, using a 4x4 block size may lead to false-positive detection. A single-pixel tampering on such a block can result in a 95 % false-positive detection rate.

The existing watermarking schemes often struggle to produce high-quality recovered images when the tampering rate exceeds 50 %. Additionally, these schemes tend to exhibit significant false-negative detections during tamper localization, leading to tamper coincidence and reduced accuracy in tamper detection. The proposed BRIWT scheme

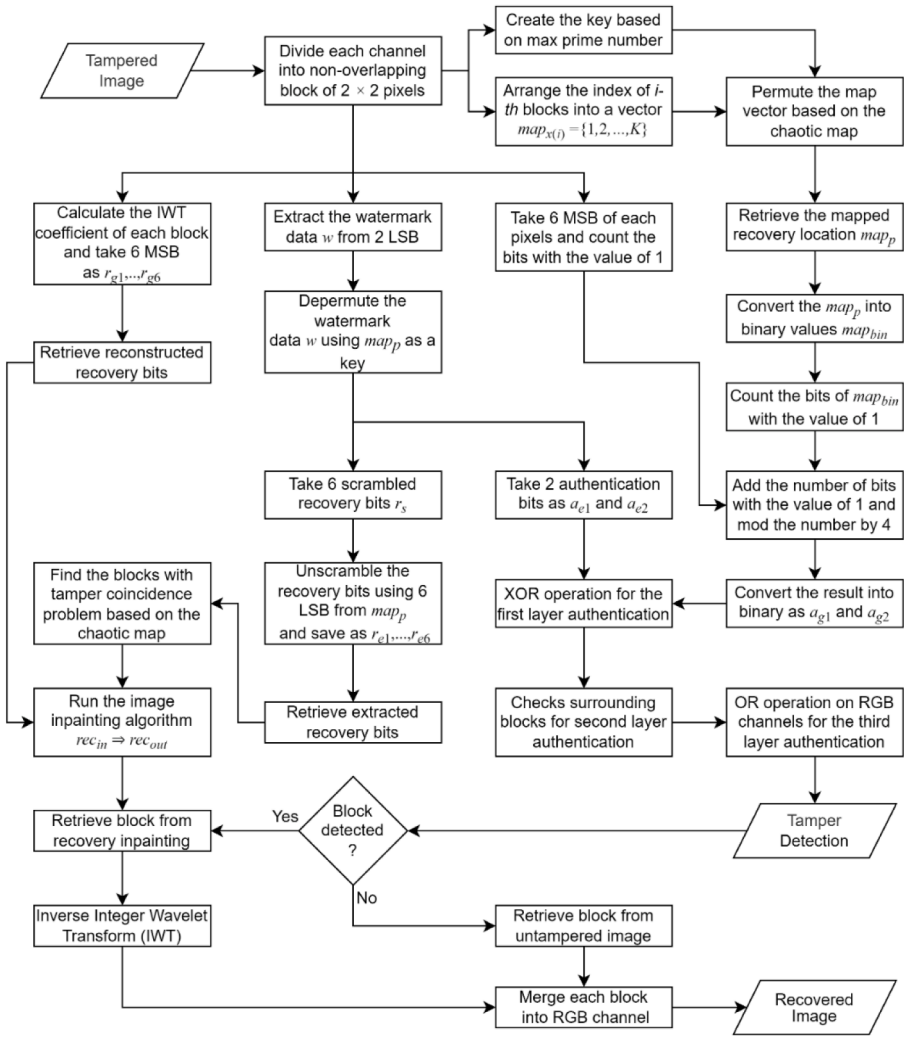

Fig. 3. The LSB adjustment in embedding watermark.

introduces several key improvements:

1. High Imperceptibility: The proposed scheme maintains high imperceptibility, achieving an average Structural Similarity Index (SSIM) value of 0.9972 and a Peak Signal-to-Noise Ratio (PSNR) value of 45.57 dB.
2. Enhanced Tamper Localization: The proposed scheme significantly improves the accuracy of tamper localization, achieving a precision rate of 0.9943 and an accuracy rate of 0.9971.
3. Innovative Image Inpainting Technique: This paper introduces a novel image inpainting technique utilizing Integer Wavelet Transform (IWT). It provides sufficient information to interpolate tampered blocks in cases of coincidence. This technique leads to a remarkable 14.2 % improvement (calculated as (21.94——19.20) / 19.20 × 100 %) under a tampering rate of 80 %.

2.3. Proposed method

The recovery data is embedded into various locations to ensure that even if the current block is tampered with, the recovery data inserted into other blocks will remain intact. The technique for embedding and extracting the watermark is further described in the paper. The process of image authentication and self-recovery encompasses both watermark embedding and watermark extraction.

2.4. Watermark embedding

The block map keeps track of the recovery bit locations. For example, the recovery bits of the first block might be located in the 20th block. Each block is mapped to a different location. The block map information is also used as a key in various stages of the watermark embedding and extraction process, so the same block map must be used for both embedding and recovery. The watermark data for each block, which includes authentication and recovery bits, is generated once the block map is created. The embedded watermark is shown in Fig. 2.

According to Fig. 2, the step-by-step of embedding process is defined by:

1. Each image is separated into small, non-overlapping 2 × 2 pixels for accurate tamper detection, while using a larger block size may result in a higher rate of false positive detections.
2. The scheme creates a vector $map_x = \{1, 2, \cdots, K\}$ to store the index of *i-th* blocks.
3. The vector is rearranged according to a specific permutation algorithm, as described in reference [25], to generate the block map data.
4. Determine the coefficient of the selected block using the lifted wavelet transform. The six most significant bits (MSB) of the LL subband values (r_1, r_2, …, r_6) are then retained as the recovery bits for the selected block.
5. To calculate the authentication bits of the selected block, first, count the number of "1″ in the most significant 6 bits of all pixels in the

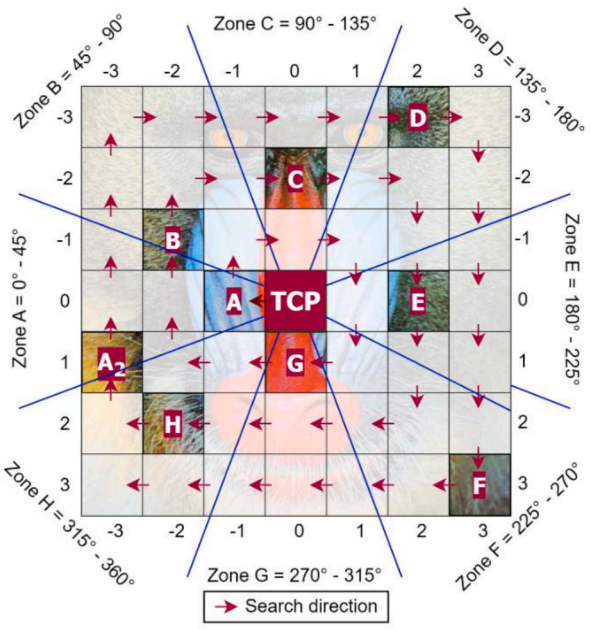

Fig. 4. The Baboon image (a) Original image (b) Difference between original image and watermarked image (c) Watermarked image.

selected block. Next, convert the block map location (*mapp*) into a binary format (*mapbin*). Then, count the number of "1" in *mapbin*. Add these two values together and divide by four. The remainder of the division is converted into a binary format with 2 bits (a_1 and a_2) which serve as the authentication bits.

6. Rearrange the recovery bits (r_1, r_2, …, r_6) of the appropriate block location into a new order and save as *rs*. Then, combine a_1, a_2, and r_s into an 8-bit watermark data. To enhance security, apply a permutation algorithm to the watermark data as outlined in reference [25].
7. To embed the watermark into the pixel, the LSB adjustment algorithm will check a set of conditions and make sure that the two least significant bits of the watermarked pixel are equal to the watermark data. For example, a pixel with an original value of 196 (11000100), and if the watermark data has the value of 3 (11). The traditional embedding method changes the two least significant bits of the original pixel value by the watermark. If the 2 least significant bits are replaced directly, the final pixel value would be 199 (11000111). In this case, 1 is subtracted from the original pixel value (196 – 1 = 195) to achieve pixel value of 195 (11000011). The visual illustration of LSB adjustment is shown in Fig. 3.
8. The process of steps 2 through 7 is repeated for every image block.

The watermark bits consist of two (2) authentication bits and six (6) recovery bits to represent each sub-block. Fig. 4 displays the original visual image, the watermarked image, and the difference between the original and the watermarked image. The watermarked image maintains a similar visual quality to the original image, as this study modifies the 2-LSB for embedding watermark bits and recovery bits. Consequently, the visual differences are not noticeable to the human visual system. The recovery bits are embedded into various sub-block locations based on a block mapping.

2.5. Tamper detection and Self-Recovery

The extracting process involves tamper detection and tamper recovery is depicted in Fig. 5.

According to Fig. 5, the block mapping is utilized to set the recovery locations. The process of extracting watermark is defined by:

1. Split an image of the tampered image into distinct blocks of 2 × 2, with no overlap.
2. Reconstruct the block map, the block map used for extraction should match the block map that was used for embedding the watermark.
3. Reconstruct the $a_{g(i)}$ and the $r_{g(i)}$ of the chosen block from the altered image.
4. Obtain the watermark data w from the 2 least significant bits.
5. Reverse the permutation of the watermark data *w* using the map_p as the key. The map_p is the placement of the recovery bits as outlined on the block map.
6. Extract the a_{e1} and a_{e2} from the depermuted data.

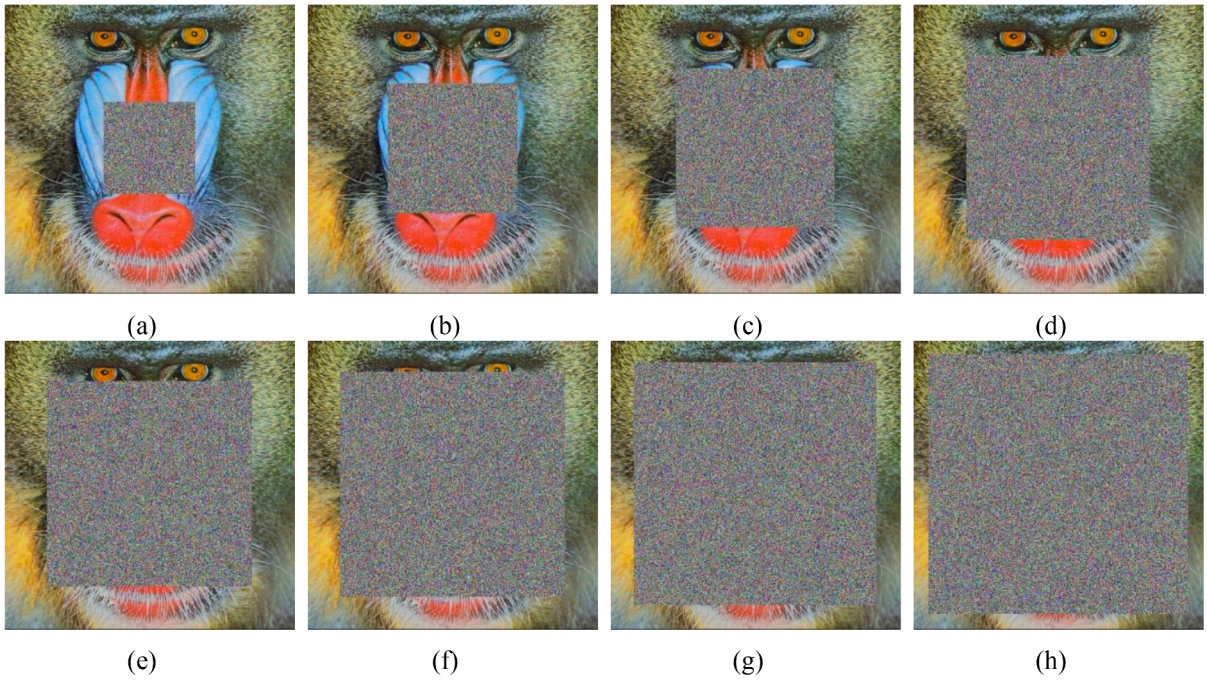

Fig. 5. The proposed BRIWT extracting watermark.

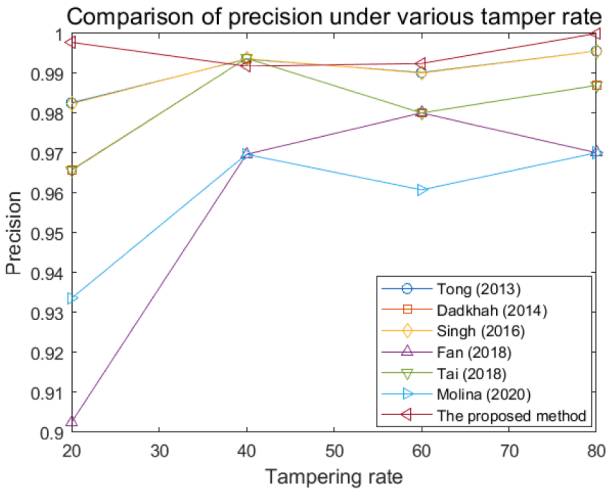

Fig. 6. Image inpainting based on Euclidean distance. Note: TCP is a tamper coincidence problem, symbols (A)-(H) are non-tamper coincidence.

7. Decrypt the watermark data to generate the.$r_{e(i)}$
8. Calculate the first level verification using the equation as follows:

$$\det_1 = logical\left(a_{g(i)} \oplus a_{e(i)}\right) \sum_{i=1}^{n} X_i^2 \quad (1)$$

where $a_{g(i)}$ is the authentication data obtained from reconstructing the watermark and $a_{e(i)}$ is the authentication data extracted from the least significant bits of the watermark. The value of 1 in a certain location of the det_1 matrix indicates that the corresponding block has been tampered with. A value of 0 could mean that the block has not been tampered with or that it has been tampered with but not detected. Using only two authentication bits results in a 25 % chance of tampering remaining undetected. Therefore, a second layer of authentication is necessary.

9. Calculate the second level of verification using the equation as follows:

$$\det_2 = count([p_1 \quad \ldots \quad p_6\ ]) > 0 \quad (2)$$

where p_1 up to p_6 represents the pairs adjacent to position i in the det_1 matrix. If both values in a pair are 1, set the corresponding p value to 1. For instance, if the location i has been altered on both the left and right, set p_1 to 1, otherwise set it to 0.

10. Apply the third level of verification to a color image using the equation as follows:

$$det_m = det_3 = det_2R \vee det_2G \vee det_2B \quad (3)$$

where det_2R represents the det_2 value for the red channel, det_2G represents the det_2 for the green channel, and det_2B represents the det_2 value for the blue channel. In case of a grayscale image, step 4 is not required.

11. Resolve the issue of tampered blocks coinciding, which occurs when both the original and recovery blocks have been altered.
12. Obtain the recovery bits of the tampered block through the IWT and then reverse the coefficients to uncover the initial block of 2 × 2 pixels.
13. Carry out steps 2 to 12 for all image blocks to generate the recovered image.

The notations (a) - (h) refer to non-TCP pixels, while TCP denotes tampered coincidence pixels. Fig. 6 illustrates eight non-TCP pixels in the image block. The inpainting algorithm is employed to address all tamper coincidences within the matrix $r_{p(i)}$. However, if the recovered block has many tamper coincidences, the algorithm will require numerous iterations to find the nearest non-TCP pixel. The image inpainting method considers eight different regions. The closest non-TCP pixel to TCP is selected if there are more non-TCP pixels in a region. In Fig. 7(a), the watermarked image displays the addition of Gaussian noise with a magnitude of 50 %. Fig. 7(b) demonstrates the effectiveness of the tamper detection mechanism. The proposed scheme successfully recovers data even in the presence of added noise, as illustrated in Fig. 7(c). However, this recovery process can lead to tamper coincidence issues, where both the recovery data and the original pixels are tampered with simultaneously. To address this problem, we employ an image inpainting technique based on eight regions, as depicted in Fig. 7(d). This approach significantly improves the visual quality after the image inpainting process.

2.6. Evaluation

The experiments evaluate the imperceptibility of the recovered image using PSNR and SSIM index. The PSNR can be computed as defined by [2627]:

$$PSNR(o,w) = 10log_{10}\left(MAX^2/MSE(o,w)\right) \quad (4)$$

$$MSE(o,w) = \frac{1}{WH}\sum_{x=1}^{M}\sum_{y=1}^{N}\left(o_{x,y} - w_{x,y}\right)^2 \quad (5)$$

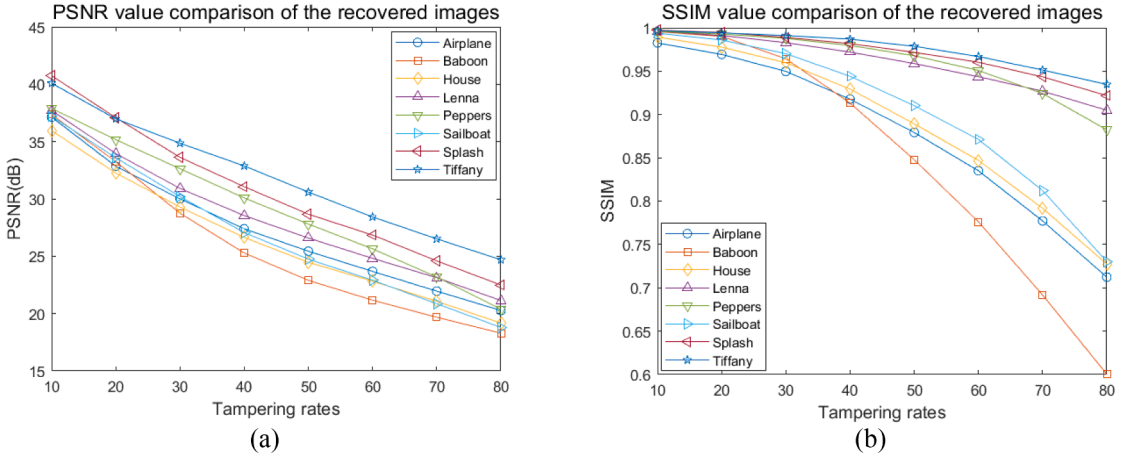

Fig. 7. The Baboon image (a) Tampered image (b) Tamper localization (c) Tamper coincidence problem (d) Recovered image.

Table 1
The comparison of PSNR value to the other existing schemes.

Image	Tong [16]	Dadkhah [17]	Singh [18]	Fan [19]	Tai [20]	Molina-Garcia [11]	AuSR1 [12]	AuSR2 [31]	BRIWT scheme
Baboon	37.90	44.14	37.90	44.12	44.14	44.64	45.70	46.06	46.37
Airplane	37.88	44.12	37.88	44.11	44.12	44.69	45.68	46.05	46.38
Lena	37.90	44.13	37.90	44.13	44.12	44.60	45.71	46.06	46.36
Sailboat	37.90	44.12	37.90	44.10	44.11	44.61	45.68	46.04	46.35
House	37.88	44.19	37.88	44.18	44.18	44.66	45.69	46.07	46.35
Tiffany	37.44	43.85	37.44	43.84	43.85	44.87	44.95	45.20	45.40
Peppers	37.79	44.06	37.79	44.06	44.06	44.54	45.54	45.87	46.16
Splash	37.84	44.08	37.84	44.08	44.09	44.47	45.57	45.93	46.22
Average	37.82	44.09	37.82	44.08	44.08	44.64	45.57	45.91	46.20

Table 2
The comparison of PSNR-HVS-M value to the other existing schemes.

Image	Tong [16]	Dadkhah [17]	Singh [18]	Fan [19]	Tai [20]	Molina-Garcia [11]	AuSR1 [12]	AuSR2 [31]	BRIWT scheme
Baboon	46.15	55.13	46.15	55.17	55.17	55.84	51.50	50.35	49.54
Airplane	41.55	50.13	41.55	50.13	50.18	49.86	49.90	49.23	48.82
Lena	41.90	50.35	41.78	50.36	50.40	50.94	49.95	49.35	48.73
Sailboat	43.76	52.61	43.72	52.62	52.72	52.18	50.65	49.84	49.24
House	42.26	50.57	42.26	50.47	50.58	51.48	50.34	49.62	48.96
Tiffany	35.61	43.17	35.61	43.17	43.15	45.56	42.93	42.95	42.85
Peppers	42.99	51.78	42.99	51.75	51.69	52.00	50.43	49.72	49.11
Splash	39.35	47.63	39.35	47.72	47.67	47.60	48.82	48.37	48.10
Average	41.69	50.17	41.67	50.17	50.19	50.68	49.32	48.68	48.17

Table 3
The comparison of SSIM value to the other existing schemes.

Image	Tong [16]	Dadkhah [17]	Singh [18]	Fan [19]	Tai [20]	Molina-Garcia [11]	AuSR1 [12]	AuSR2 [31]	BRIWT scheme
Baboon	0.9763	0.9941	0.9763	0.9941	0.9941	0.9947	0.9990	0.9991	0.9992
Airplane	0.9194	0.9782	0.9194	0.9781	0.9781	0.9812	0.9889	0.9901	0.9914
Lena	0.9307	0.9820	0.9307	0.9820	0.9820	0.9840	0.9993	0.9994	0.9995
Sailboat	0.9494	0.9868	0.9493	0.9867	0.9868	0.9884	0.9980	0.9982	0.9984
House	0.9319	0.9815	0.9319	0.9815	0.9815	0.9834	0.9967	0.9970	0.9974
Tiffany	0.9246	0.9806	0.9246	0.9804	0.9805	0.9846	0.9985	0.9986	0.9987
Peppers	0.9234	0.9791	0.9234	0.9791	0.9791	0.9816	0.9991	0.9992	0.9992
Splash	0.8942	0.9695	0.8942	0.9695	0.9696	0.9737	0.9983	0.9985	0.9987
Average	0.9312	0.9815	0.9312	0.9814	0.9815	0.9840	0.9972	0.9975	0.9978

where $o_{x,y}$ and $w_{x,y}$ are the cover image and watermarked image, and x, y is the image's pixel coordinate. PSNR is represented in decibel units, and SSIM, developed by Wang et al. [28], is a more recent method for determining image quality. SSIM compares two images and is closely aligned with how the human eye perceives image quality [29]. The calculation for SSIM is as follows [30]:

$$SSIM(o,w) = l(o,w)c(o,w)s(o,w) \tag{6}$$

$$l(o,w) = \frac{2\mu_o\mu_w + C_1}{\mu_o^2 + \mu_w^2 + C_1} \tag{7}$$

$$c(o,w) = \frac{2\sigma_o\sigma_w + C_2}{\sigma_o^2 + \sigma_w^2 + C_2} \tag{8}$$

$$s(o,w) = \frac{\sigma_{ow} + C_3}{\sigma_o\sigma_w + C_3} \tag{9}$$

The SSIM is calculated using the luminance, the contrast, and the structure function. Tamper detection is evaluated using four metrics: TPR, FNR, FPR, TNR, precision, F1-score, and accuracy as given by:

$$TPR = recall = \frac{TP}{TP + FN} \tag{10}$$

$$FNR = \frac{FN}{TP + FN} \tag{11}$$

$$FPR = \frac{FP}{FP + TN} \tag{12}$$

$$TNR = \frac{TN}{FP + TN} \tag{13}$$

$$precision = \frac{TP}{TP + FP} \tag{14}$$

$$F1 - score = 2 \times \frac{precision \times recall}{precision + recall} \tag{15}$$

$$accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{16}$$

where TPR is computed as the correctly identified watermarked images divided by the sum of true positives (correctly identified watermarked images) and false negatives (watermarked images that were not detected). A high true-positive rate indicates that the watermark detector can correctly identify a significant proportion of watermarked images. FNR is calculated as the number of false negatives (watermarked images that were not detected) divided by the sum of true positives (correctly identified watermarked images) and false negatives. A high false-negative rate indicates that the classifier is missing many true positives and failing to detect the watermarks on the images. Precision is a measure of the proportion of positive instances (predicted tampered images). A high precision indicates that the watermark detector can

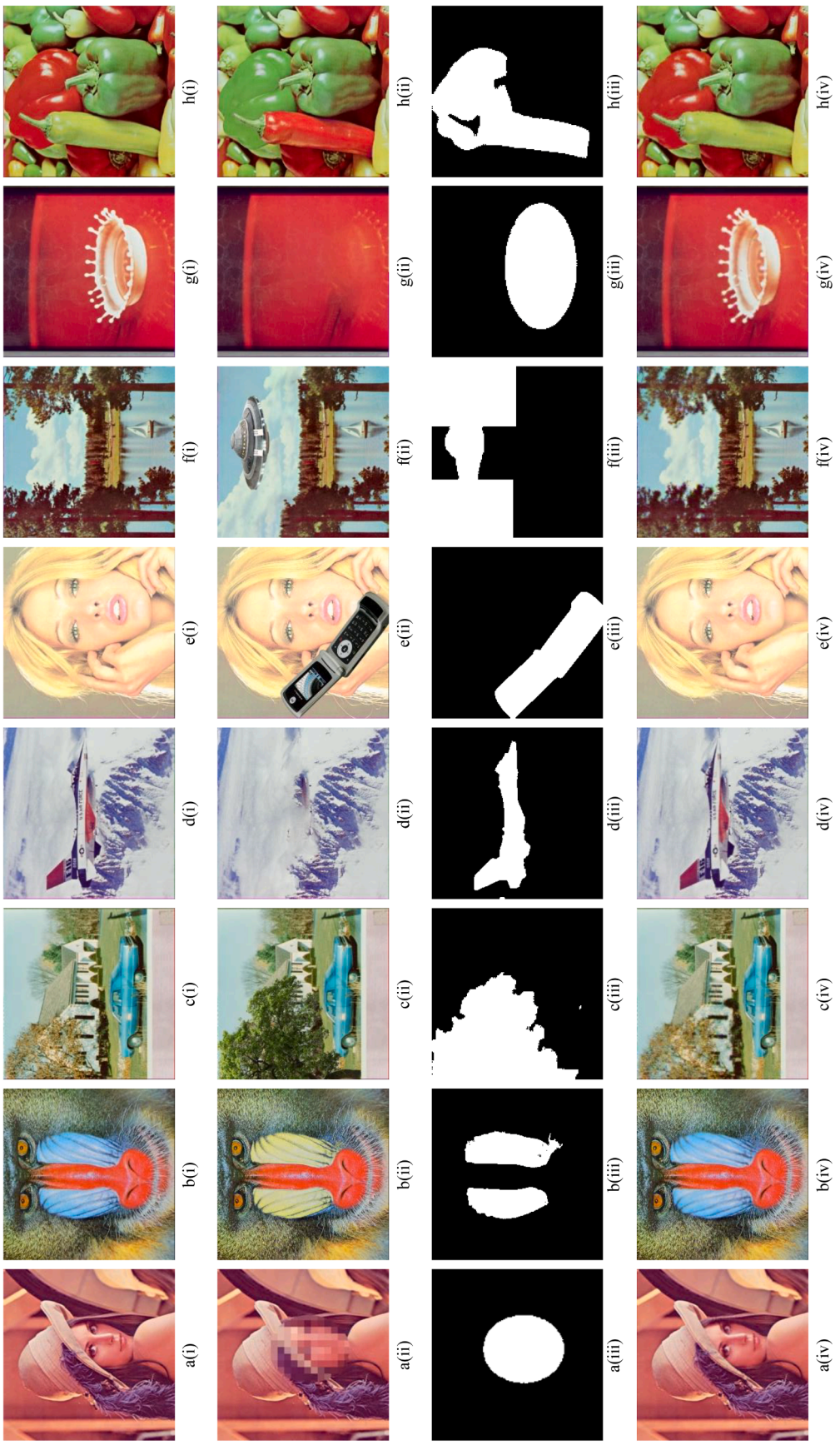

Fig. 8. Watermarked image under tampering rate: (a) 10%, (b) 20%, (c) 30%, (d) 40%, (e) 50%, (f) 60%, (g) 70%, (h) 80%.

Table 4
The precision, F1-score, and accuracy values of the tamper localization under various tampering rates.

Tampering Rates	TPR	FNR	FPR	TNR	Precision	F1-Score	Accuracy
10	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	1.0000
20	1.0000	0.0000	0.0022	0.9978	0.9978	0.9989	0.9989
30	1.0000	0.0000	0.0061	0.9939	0.9939	0.9970	0.9969
40	1.0000	0.0000	0.0083	0.9917	0.9918	0.9959	0.9959
50	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	1.0000
60	1.0000	0.0000	0.0076	0.9924	0.9925	0.9962	0.9962
70	1.0000	0.0000	0.0217	0.9783	0.9787	0.9893	0.9891
80	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	1.0000
Average	1.0000	0.0000	0.0057	0.9943	0.9943	0.9972	0.9971

Table 5
The comparison of the precision value for tamper localization to the other existing schemes.

Tampering Rates	Tong [16]	Singh [18]	Fan [19]	Tai [20]	Molina-Garcia [11]	AuSR1 [12]	AuSR2 [31]	BRIWT Scheme
10	0.9855	0.9855	0.8007	0.9670	0.9152	1.0000	0.9986	1.0000
20	1.0000	1.0000	0.9210	0.9855	0.9580	0.9978	0.9934	0.9978
30	1.0000	1.0000	0.9144	0.9903	0.9716	0.9939	0.9909	0.9939
40	0.9963	0.9963	0.9483	0.9939	0.9797	0.9918	0.9959	0.9918
50	1.0000	1.0000	1.0000	1.0000	0.9884	1.0000	0.9890	1.0000
60	0.9975	0.9975	0.9601	0.9943	0.9848	0.9925	0.9925	0.9925
70	1.0000	1.0000	0.9748	0.9958	0.9876	0.9787	0.9785	0.9787
80	1.0000	1.0000	0.9659	0.9963	0.9891	1.0000	0.9500	1.0000
Average	0.9974	0.9974	0.9357	0.9904	0.9718	0.9943	0.9861	0.9943

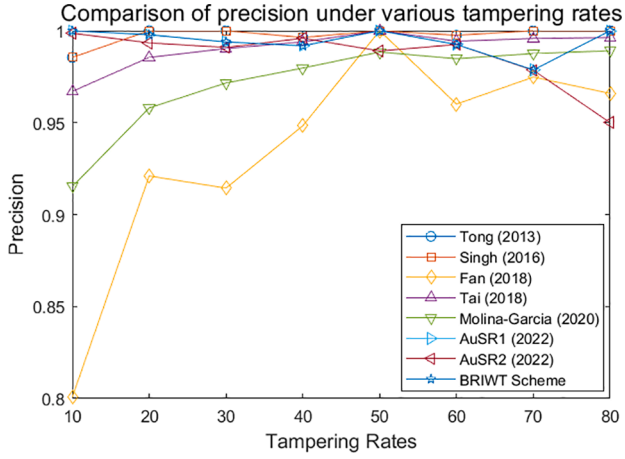

Fig. 9. Comparison of precision value under different tampering levels.

correctly identify a significant proportion of watermarked images among all the images it predicts as watermarked. The F1-score is a valuable metric when balancing precision and recall is crucial, as in the case of tamper detection. It helps avoid both false positives (non-tampered images incorrectly identified as tampered) and false negatives (tampered images not detected). Accuracy is the ratio of correctly identified tampered images, whether they are tampered or not. A high accuracy indicates that the proposed scheme can correctly identify a significant proportion of tampered and non-tampered locations.

3. Experimental results

The proposed watermarking scheme was tested using eight color images from the USC-SIPI database, each with a size of 512 x 512 pixels. To improve tamper detection and precision, the cover image was divided into small 2 × 2-pixel blocks. Information was embedded into the two least significant bits to strike a balance between tampering detection accuracy and recovery data storage capacity. An LSB adjustment algorithm was also applied to enhance imperceptibility. The quality of the watermarked image was assessed using both PSNR and SSIM measurements, comparing it to the original cover image. Tables 1,

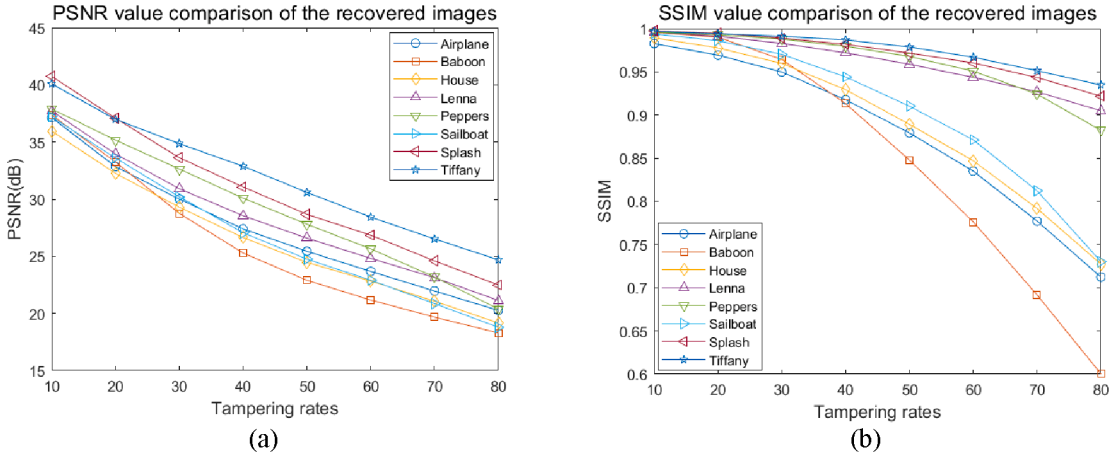

Fig. 10. The comparison of PSNR and SSIM value for different images against different tampering rates (a) PSNR (b) SSIM.

Fig. 11. (i) Original (ii) Tampered (iii) Tamper localization (iv) Recovered image.

Table 6
Comparison of the PSNR value of the proposed scheme and the existing scheme.

TR	Tong [16]	Dadkhah [17]	Singh [18]	Fan [19]	Tai [20]	Molina-Garcia [11]	AuSR1 [12]	AuSR2 [31]	BRIWT scheme
10	34.2	22.51	26.55	31.47	25.89	37.34	37.96	**38.11**	38.05
20	25.77	17.32	21.47	28.36	20.57	33.98	34.65	34.21	**34.76**
30	21.04	14.52	18.27	21.62	17.43	31.28	31.79	31.10	**31.88**
40	17.26	12.64	15.96	15.79	15.21	28.47	29.48	28.63	**29.55**
50	14.29	11.40	14.16	15.69	13.54	26.00	27.64	26.43	**27.68**
60	11.84	10.39	12.59	11.57	12.01	23.51	25.72	24.60	**25.83**
70	9.82	9.61	11.29	11.57	10.80	21.23	23.80	22.66	**23.97**
80	8.11	9.03	10.23	8.10	9.81	19.20	21.63	20.64	**21.92**

Table 7
Comparison of the PSNR-HVS-M value of the proposed scheme and the existing scheme.

TR	Tong [16]	Dadkhah [17]	Singh [18]	Fan [19]	Tai [20]	Molina-Garcia [11]	AuSR1 [12]	AuSR2 [31]	BRIWT scheme
10	34.20	18.32	23.39	30.53	23.18	36.39	**39.59**	39.35	39.53
20	25.77	13.19	18.20	27.49	17.77	32.34	34.94	33.94	**35.12**
30	21.04	10.51	14.91	20.08	14.74	29.15	**31.15**	30.00	31.09
40	17.26	8.75	12.61	14.07	12.65	25.64	28.11	27.10	**28.31**
50	14.29	7.60	10.76	13.95	11.13	22.70	25.72	24.49	**25.90**
60	11.84	6.67	9.18	9.89	9.76	19.84	23.35	22.38	**23.60**
70	9.82	5.92	7.80	9.89	8.68	17.31	21.18	20.13	**21.46**
80	8.11	5.36	6.69	6.41	7.82	15.11	18.50	17.62	**18.75**

Table 8
Comparison of the SSIM value of the proposed scheme and the existing scheme.

TR	Tong [16]	Dadkhah [17]	Singh [18]	Fan [19]	Tai [20]	Molina-Garcia [11]	AuSR1 [12]	AuSR2 [31]	BRIWT scheme
10	0.9733	0.9131	0.9290	0.9731	0.9384	0.9714	0.9928	**0.9935**	0.9934
20	0.9171	0.7983	0.8310	0.9502	0.8443	0.9390	0.9864	0.9864	**0.9872**
30	0.8282	0.6855	0.7257	0.8875	0.7364	0.8977	0.9742	0.9734	**0.9751**
40	0.715	0.5731	0.6215	0.7230	0.6226	0.8368	0.9555	0.9534	**0.9567**
50	0.5849	0.4704	0.5139	0.7202	0.5135	0.7571	0.9339	0.9255	**0.9355**
60	0.452	0.3586	0.3984	0.4249	0.3899	0.6460	0.9059	0.8932	**0.9078**
70	0.3233	0.2506	0.2855	0.4249	0.2744	0.5157	0.8705	0.8490	**0.8737**
80	0.2042	0.1511	0.1799	0.0094	0.1655	0.3958	0.8219	0.7937	**0.8280**

2 and 3 provide a comparison of this method with existing methods.

The proposed method outperformed existing methods when evaluated using PSNR and SSIM values. However, the evaluation with PSNR-HVS-M produced slightly lower results, achieving 48.17 dB, in comparison to other schemes. This trend was consistent when assessing the average performance across all test images. the image known as "Tiffany" exhibited the lowest quality among the watermarked images due to its limited texture. Images with less texture are more affected by the high-frequency data added during the embedding process. Additionally, the PSNR values for embedding the watermark into 1, 2, and 3 LSBs were 51 dB, 44 dB, and 37 dB, respectively. The proposed method, which includes an LSB adjustment algorithm, effectively maintains watermarked imperceptibility with an SSIM score of approximately 0.9978. Furthermore, while most previous schemes only consider grayscale images, this method can be applied to both grayscale and color images, demonstrating its versatility and effectiveness.

3.1. Image authentication

The proposed scheme was evaluated against collage attacks with different tamper attack sizes. This experiment encompasses two attack scenarios. The first scenario is a regular attack, involving the addition of noise to the central region of the images, ranging from 10 % to 80 %, as depicted in Fig. 8. This attack replicates the one presented by Molina-Garcia [15], and he also compared tamper detection and recovery based on this attack. The experimental results for tamper localization, including precision, F1-score, and accuracy values, are provided in Table 4. Furthermore, Table 5 offers a comparison of precision results with other existing schemes for tamper localization. A visual representation of the precision values obtained from the existing methods is shown in Fig. 9. The second scenario involves an irregular attack, where the watermarked images are altered, as illustrated in Fig. 11. This simulates a real-life attack that may occur in a communication channel.

According to Fig. 9, the proposed scheme demonstrates high precision in tamper detection compared to existing schemes at tampering rates of 20 %, 60 %, and 80 %. The proposed scheme employs several authentication approaches to enhance the precision of tamper detection. As discussed earlier, the proposed scheme utilizes three authentication layers to increase true-positive detections. However, this approach may reduce the capacity for storing recovery bits. Irregular attacks were applied to the watermarked image, and the results of tamper localization and recovery are illustrated in Fig. 11.

3.2. Self-recovery

Tamper localization directly affects the quality of the recovered image. High precision in tamper localization results can lead to a high-quality recovered image. The embedded recovery bits are stored in separate locations based on block mapping. However, both the original embedding location and the recovery location may be tampered with, leading to what is referred to as a tamper coincidence problem. When tamper coincidence occurs, the quality of image recovery is compromised, resulting in poor quality and artifacts in the image. This issue can be addressed by employing a multi-layer recovery approach and image inpainting.

A multi-layer recovery approach requires space to store the recovery data, which can impact the imperceptibility of the watermarked image and the precision of tamper localization performance. Image inpainting techniques can interpolate the tampered block based on the surrounding non-tampered region. However, when the tampered region is large, it may produce a blocking effect. The proposed inpainting technique is based on eight regions, as demonstrated in Fig. 6. Experimental results demonstrate that it produces a high-quality recovered image, as shown in Fig. 10.

The Tiffany image produced the highest SSIM value when compared to other tampered images due to its smoothness and lack of complex textures. The use of IWT and image inpainting techniques improves the recovery process, resulting in a high PSNR and SSIM value compared to the original image. In contrast, the Baboon image, which has more texture, exhibits the worst recovery quality. The proposed method, utilizing Integer Wavelet Transform and image inpainting, accurately authenticates tampered images and generates high-quality recovered images, even in irregular attack scenarios, as demonstrated in Fig. 11. The image inpainting technique effectively recovers tampered areas with fine details when the tampering rate is low. However, as the tampering rate increases, the ability to recover tampered areas decreases. Despite this, our proposed method still outperforms existing methods, as shown in Tables 6, 7, and 8.

According to Tables 6, 7, and 8, our method exhibits significant improvement in the quality of the recovered image across tampering rates ranging from 10 % to 80 %. Even at a high tampering rate of 80 %, our method maintains a PSNR of 21.92 dB, surpassing the previous method's PSNR of 20 dB. Additionally, our method achieves a superior SSIM of 0.8280 under an 80 % tampering rate, whereas the previous method only achieves an SSIM of 0.3958. This demonstrates the substantial performance enhancement offered by our proposed method.

The limitation of this study lies in the limited embedding capacity for storing recovery bits. Since the proposed BRIWT scheme embeds authentication bits and recovery bits in the 2-LSB of 2x2 block pixels, the BRIWT scheme can only store 6 recovery bits and 2 authentication bits. Increasing the embedding capacity by increasing the block size can store more recovery data, but it may result in a slight decrease in the accuracy and precision of tamper localization. The minimum block size of 2x2 pixels, which embeds 2-LSB and 2 authentication bits, has a significant impact on the high precision and accuracy of tamper localization. The embedding capacity of 2-LSB in the 2x2 block pixels does not significantly affect the watermarked quality. However, due to the limited number of recovery bits available for embedding in the 2x2 block pixels, it can lead to a decrease in the quality of the recovered image. Having a higher capacity for recovery bits can improve the quality of the recovered image from a tampered image, but it will also decrease the accuracy and precision of tamper localization at the same time.

4. Conclusions

This paper presents a blind recovery technique using integer wavelet transform (BRIWT) scheme for recovering images against various tampering rate attacks. The proposed method inserts check bits and recovery data into the two least significant bits (LSB). The multi-layer authentication process effectively detects tampered regions. Recovery bits are randomly embedded into different block locations and are generated using the Integer Wavelet Transform (IWT) in the LL subband. Our scheme also introduces an image inpainting technique for replacing tampered pixels by considering the nearest eight regions of untampered pixels. The results demonstrate that this method produces high-quality watermarked images using the LSB adjustment algorithm. The proposed scheme was able to produce superior recovered images against regular attacks. For future studies, given that the proposed BRIWT scheme can produce a high-quality watermarked image while considerably improving tamper localization accuracy, it has the potential to be used in video watermarking. This method can produce watermarked videos with great imperceptibility while also allowing for accurate validation of the video content's integrity.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported by the Ministry of Higher Education for providing financial support under Fundamental Research Grant Scheme (FRGS), No. FRGS/1/2022/ICT04/UMP/02/2 (University reference RDU220133).

References

[1] F. Ernawan, D. Ariatmanto, A recent survey on image watermarking using scaling factor techniques for copyright protection, Multimed. Tools Appl. 82 (18) (2023) 27123–27163, https://doi.org/10.1007/s11042-023-14447-5.
[2] G. Suresh, V.L. Narla, D.P. Gangwar, A.K. Sahu, False-Positive-Free SVD Based Audio Watermarking with Integer Wavelet Transform, Circuits Syst Signal Process 41 (9) (2022) 5108–5133, https://doi.org/10.1007/s00034-022-02023-5.
[3] L. Chen, J. Zhao, Contourlet-based image and video watermarking robust to geometric attacks and compressions, Multimed. Tools Appl. 77 (6) (2018) 7187–7204, https://doi.org/10.1007/s11042-017-4628-7.
[4] S. Sharma, J.J. Zou, G. Fang, P. Shukla, W. Cai, A review of image watermarking for identity protection and verification, Multimed. Tools Appl. (2023), https://doi.org/10.1007/s11042-023-16843-3.
[5] N. Agarwal, A.K. Singh, P.K. Singh, Survey of robust and imperceptible watermarking, Multimed. Tools Appl. 78 (7) (2019) 8603–8633, https://doi.org/10.1007/s11042-018-7128-5.
[6] A.K. Sahu, M. Sahu, P. Patro, G. Sahu, S.R. Nayak, Dual image-based reversible fragile watermarking scheme for tamper detection and localization, Pattern Anal. Appl. 26 (2) (2023) 571–590, https://doi.org/10.1007/s10044-022-01104-0.
[7] A.K. Sahu, A logistic map based blind and fragile watermarking for tamper detection and localization in images, J. Ambient Intell. Hum. Comput. 13 (8) (2022) 3869–3881, https://doi.org/10.1007/s12652-021-03365-9.
[8] A.K. Sahu, M. Hassaballah, R.S. Rao, G. Suresh, Logistic-map based fragile image watermarking scheme for tamper detection and localization, Multimed. Tools Appl. 82 (16) (2023) 24069–24100, https://doi.org/10.1007/s11042-022-13630-4.
[9] H. Rhayma, A. Makhloufi, H. Hamam, and A. Ben Hamida, "Semi-fragile watermarking scheme based on perceptual hash function (PHF) for image tampering detection," *Multimed. Tools Appl. 2021 8017*, vol. 80, no. 17, pp. 26813–26832, May 2021, doi: 10.1007/S11042-021-10886-0.
[10] N. Sivasubramanian, G. Konganathan, "A novel semi fragile watermarking technique for tamper detection and recovery using IWT and DCT," *Comput. 2020 1026*, vol. 102, no. 6, pp. 1365–1384, Feb. 2020, doi: 10.1007/S00607-020-00797-7.
[11] J. Molina-Garcia, B.P. Garcia-Salgado, V. Ponomaryov, R. Reyes-Reyes, S. Sadovnychiy, C. Cruz-Ramos, An effective fragile watermarking scheme for color image tampering detection and self-recovery, Signal Process. Image Commun. 81 (2020), 115725, https://doi.org/10.1016/j.image.2019.115725.
[12] A. Aminuddin, F. Ernawan, AuSR1: Authentication and self-recovery using a new image inpainting technique with LSB shifting in fragile image watermarking, J. King Saud Univ. - Comput. Inf. Sci. (2022), https://doi.org/10.1016/J.JKSUCI.2022.02.009.
[13] B. Bolourian Haghighi, A. H. Taherinia, A. H. Mohajerzadeh, "TRLG: Fragile blind quad watermarking for image tamper detection and recovery by providing compact digests with optimized quality using LWT and GA," *Inf. Sci. (Ny).*, vol. 486, pp. 204–230, Jun. 2019, doi: 10.1016/j.ins.2019.02.055.
[14] F. Ernawan, D. Ariatmanto, Image watermarking based on integer wavelet transform-singular value decomposition with variance pixels, Int. J. Electr. Computer Eng. 9 (3) (2019) 2185–2195, https://doi.org/10.11591/ijece.v9i3.pp2185-2195.
[15] F. Ernawan, A. Aminuddin, D. N. E. Phon, E. A. Alsheikh, and A. Wibowo, "Self-Recovery in Fragile Image Watermarking Using Integer Wavelet Transform," Nov. 2022, pp. 21–25. doi: 10.1109/icsima55652.2022.9929127.
[16] X. Tong, Y. Liu, M. Zhang, Y. Chen, A novel chaos-based fragile watermarking for image tampering detection and self-recovery, Signal Process. Image Commun. 28 (3) (2013) 301–308, https://doi.org/10.1016/j.image.2012.12.003.
[17] S. Dadkhah, A. Abd Manaf, Y. Hori, A. Ella Hassanien, S. Sadeghi, An effective SVD-based image tampering detection and self-recovery using active watermarking, Signal Process. Image Commun. 29 (10) (2014) 1197–1210, https://doi.org/10.1016/J.IMAGE.2014.09.001.
[18] D. Singh, S.K. Singh, Effective self-embedding watermarking scheme for image tampered detection and localization with recovery capability, J. Vis. Commun. Image Represent. 38 (2016) 775–789, https://doi.org/10.1016/J.JVCIR.2016.04.023.
[19] M.Q. Fan, H.X. Wang, An enhanced fragile watermarking scheme to digital image protection and self-recovery, Signal Process. Image Commun. 66 (2018) 19–29, https://doi.org/10.1016/J.IMAGE.2018.04.003.
[20] W.L. Tai, Z.J. Liao, Image self-recovery with watermark self-embedding, Signal Process. Image Commun. 65 (2018) 11–25, https://doi.org/10.1016/J.IMAGE.2018.03.011.
[21] D. Sarkar, S. Palit, S. Som, K.N. Dey, Large scale image tamper detection and restoration, Multimed. Tools Appl. 79 (25–26) (Jul. 2020) 17761–17791, https://doi.org/10.1007/s11042-020-08669-0.
[22] T. Liu, X. Yuan, A dual-tamper-detection method for digital image authentication and content self-recovery, Multimed. Tools Appl. 80 (19) (Aug. 2021) 29805–29826, https://doi.org/10.1007/s11042-021-11179-2.
[23] S.N.V.J. Devi Kosuru, G. Swain, N. Kumar, A. Pradhan, Image tamper detection and correction using Merkle tree and remainder value differencing, Optik (stuttg) 261 (2022) Jul, https://doi.org/10.1016/j.ijleo.2022.169212.
[24] C.C. Lin, X. Liu, J.J. Zhou, C.Y. Tang, An image authentication and recovery scheme based on turtle Shell algorithm and AMBTC-compression, Multimed. Tools Appl. 81 (27) (2022) 39431–39452, https://doi.org/10.1007/s11042-022-12995-w.
[25] C. Jefferson, M. Pfeiffer, W.A. Wilson, R. Waldecker, Permutation group algorithms based on directed graphs, J. Algebr. 585 (2021) 723–758, https://doi.org/10.1016/J.JALGEBRA.2021.06.015.
[26] F. Ernawan, Tchebichef image watermarking along the edge using YCoCg-R color space for copyright protection, Int J. Electr. Computer Eng. 9 (3) (2019) 1850–1860, https://doi.org/10.11591/ijece.v9i3.pp1850-1860.
[27] D. Ariatmanto, F. Ernawan, Adaptive scaling factors based on the impact of selected DCT coefficients for image watermarking, J. King Saud Univ. - Computer Inform. Sci. 34 (3) (2022) 605–614, https://doi.org/10.1016/j.jksuci.2020.02.005.
[28] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: From error visibility to structural similarity, IEEE Trans. Image Process. 13 (4) (Apr. 2004) 600–612, https://doi.org/10.1109/TIP.2003.819861.
[29] A. Horé and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *Proceedings - International Conference on Pattern Recognition*, 2010, pp. 2366–2369, doi: 10.1109/ICPR.2010.579.
[30] F. Ernawan, Robust image watermarking based on psychovisual threshold, J. ICT Res. Appl. 10 (3) (2016) 228–242, https://doi.org/10.5614/itbj.ict.res.appl.2016.10.3.3.
[31] A. Aminuddin, F. Ernawan, AuSR2: Image watermarking technique for authentication and self-recovery with image texture preservation, Comput. Electr. Eng. 102 (2022), https://doi.org/10.1016/j.compeleceng.2022.108207.