Engineering Science and Technology, an International Journal 56 (2024) 101790

Contents lists available at ScienceDirect

Engineering Science and Technology, an International Journal

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

TCBR and TCBD: Evaluation metrics for tamper coincidence problem in fragile image watermarking

Afrig Aminuddin [a,b], Ferda Ernawan [a,*], Danakorn Nincarean [a], Agit Amrullah [a,b], Dhani Ariatmanto [b]

[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:
Block mapping
Image authentication
Image watermarking
Self-recovery
Tamper coincidence problem
Tamper localization

ABSTRACT

This paper proposed two evaluation metrics of the tamper coincidence in a block map design for image watermarking. These evaluation metrics are called Tamper Coincidence Block Ratio (TCBR) and Tamper Coincidence Block Density (TCBD). A tamper coincidence occurred in image authentication and self-recovery when the recovery data and the original block location were tampered with simultaneously. A high tamper coincidence limits image inpainting's capability to recover the region, leading to an imprecise recovered image. The ratio and density of the tamper coincidence may significantly affect the final recovered image quality. Previously, researchers mentioned the tamper coincidence in their experiment but did not evaluate it with any metrics. They evaluated the robustness of their technique based on the final recovered image quality using the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM). Tamper coincidences are primarily affected by the block map design implemented by the researcher. Thus, TCBR and TCBD provide valuable insight into the block map design's effectiveness in preventing tamper coincidence. The experimental result shows that the TCBR and TCBD values are inversely proportional to the recovered image quality. A high TCBR and TCBD value leads to low recovered image quality. Therefore, this paper will help the researchers design an effective block map by minimizing the TCBR and TCBD values to obtain the highest recovered image quality.

1. Introduction

In the digital era, the authenticity and integrity of images are critical, requiring image authentication and self-recovery techniques to be implemented. Image authentication verifies an image's originality and integrity, employing techniques of image watermarking to detect unauthorized tampering [1]. Self-recovery extends this by localizing the tampering area and restoring the image to its original image. In image authentication and self-recovery based on image watermarking, the authentication and recovery data are embedded as a watermark into the original image, which outputs a watermarked image. The watermarked image is then considered secure and can be transmitted through the internet [2]. Once the intended recipient accepts the image, they can verify its authenticity. In addition, the recipient can also recover the image to its original state if it was tampered with during the transmission process. The recovered image quality is affected by the availability of the recovery data on the tampered image. If a large region of the image is tampered with, a large part of the recovery data will be missing, leading to a low-quality recovered image. Missing recovery data is also called the tamper coincidence [3].

In image authentication and self-recovery, two types of watermark data are embedded into the cover image: authentication and recovery data. In the embedding process, the authentication data is embedded into its original location, while the recovery data is mapped and embedded into another location within the image. The researcher calls the map of the recovery data the block map [4]. Designing an efficient block map with the tamper coincidence in mind is essential. When the recovery data is embedded near the original block location, the recovery data may be lost during tampering as the attacker modifies both the original block of the image and the recovery data [5]. Therefore, the block map should be designed to avoid such an attack by ensuring the recovery data is embedded into the furthest location from the original block location.

In the previous research, image authentication and self-recovery

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

https://doi.org/10.1016/j.jestch.2024.101790
Received 14 May 2024; Received in revised form 14 July 2024; Accepted 1 August 2024

framework are only evaluated in 3 stages: watermarked image quality, tamper localization accuracy, and recovered image quality. The watermarked and recovered image is compared to the original image to find its PSNR and SSIM value. Meanwhile, the tamper localization accuracy compares the ground truth with the tamper localization image output [6]. The ground truth can be obtained by comparing the watermarked image with the tampered image in every pixel. The researchers only evaluated their framework on the recovered image quality based on the post-processing stage during the self-recovery process, while the tamper coincidence caused by the block map design was not thoroughly evaluated [7].

This paper proposed two evaluation metrics for the block map design in image authentication and self-recovery framework. The evaluation metrics are Tamper Coincidence Block Ratio (TCBR) and Tamper Coincidence Block Density (TCBD). TCBR measures the ratio between the tamper coincidence block and the total number of blocks within the image. TCBR will provide insight into how efficiently the block map is designed to reduce the total tamper coincidence block. Meanwhile, TCBD calculates the density of the tamper coincidence block. A high concentration of tamper coincidence block will limit the post-processing stage of the self-recovery process.

The motivation of the paper is to address the challenge of tamper coincidence in image watermarking and to propose new evaluation metrics for assessing the effectiveness of block map designs in preventing such tampering. The paper aims to provide insights into designing block maps that minimize tamper coincidence, thereby enhancing the quality of recovered images after tampering attacks. The contribution of this paper in advancing the field of image authentication and self-recovery is highlighted as follows:

1) New evaluation metrics: Introduction of two new metrics, TCBR and TCBD, for evaluating the effectiveness of block map designs in image watermarking
2) Insight into block map design: The study provides valuable insights into designing block maps that minimize TCBR and TCBD values.
3) Correlation analysis: Analysis of the correlation between TCBR, TCBD, and traditional image quality assessments like PSNR and SSIM, highlighting the impact of tamper coincidence on recovered image quality.

These contributions aim to improve the design of block maps in image watermarking and enhance the self-recovery capability of watermarked images. The rest of this paper is organized as follows: Section II discusses the existing schemes with the tamper coincidence. Section III provides the proposed method to measure the TCBR and TCBD. Section IV shows the experimental results and the correlation between the existing image quality assessment and the proposed method. Finally, Section V concludes this research.

2. Related works

Researchers have studied the problem of tamper coincidence since image authentication supports self-recovery [8]. The self-recovery technique requires preliminary data that was previously embedded into the image to recover the tampered region of the image. These preliminary or recovery data are embedded into the image as a watermark. In the watermark embedding process, the algorithm divides the image into blocks with a specified size. This block division enables the algorithm to store more data for recovery instead of pixel-basis embedding, which can only store 2 bits of data for 2-LSB embedding. To store the recovery data, the algorithm should at least store 8 bits representing the average value of the image block, which requires 4 pixels in a single block. This recovery block is commonly embedded into another block location so that if the original block is tampered with, the recovery data can be extracted to recover the original block [9]. The tamper coincidence occurs when the recovery data is corrupted due to a large tampering area, which attacks image blocks and their recovery blocks. There are several techniques to reduce the tamper coincidence: block mapping, multiple recoveries, and image inpainting.

2.1. Block mapping

The most straightforward technique to prevent the tamper coincidence is to embed the recovery data into another block within an image. Before the recovery data is embedded, the algorithm should decide the embedding location of the recovery data. The map showing the target embedding location of each block is called a block map. The technique employed to design the block map is called a block mapping technique. The most common technique for block mapping is the random block mapping technique. It generates the block map using the Pseudo Random Number Generator (PRNG). The recovery data is embedded into another location based on the random number provided by the PRNG. The randomness of this number relies on the internal PRNG implementation such as chaotic map [10], logistic chaotic map [11], and binary map [12]. Several techniques have been developed to embed the recovery data using this block mapping technique.

Singh and Singh [13] presented a Discrete Cosine Transform (DCT) based on a self-recoverable fragile watermarking scheme for image tampering detection and localization with recovery capability. The scheme employed a two-level hierarchical tamper detection mechanism. This mechanism enhanced the accuracy of tamper localization and ensured a high probability of detection for tampered blocks. The paper solved the tamper coincidence by embedding recovery bits in a mapped block instead of the block itself using a random block mapping technique. Regarding recovery quality, the paper highlighted using small block sizes, a smoothing function, and DCT coefficients for recovery bits.

Aminuddin and Ernawan [14] presented AuSR1, a color image authentication scheme using blind fragile image watermarking for tamper detection and self-recovery. It divided the image into 2×2 pixel blocks, embedded authentication data in the original location, and placed recovery data in a distant location based on a block mapping algorithm using PRNG. The watermark data was embedded into the 2-LSB to maintain high-quality recovered images under tampering attacks. The scheme included a permutation algorithm for security and a three-layer authentication algorithm for high detection rates. Experimental results showed high PSNR and SSIM values for watermarked and recovered images, indicating the scheme's effectiveness in detecting tampered areas and recovering images.

Aminuddin and Ernawan [15] presented AuSR3, a novel block mapping technique for image authentication and self-recovery, addressing the tamper coincidence problem where both the original block and its recovery data were tampered with, rendering recovery impossible. To minimize this issue, AuSR3 embedded recovery data at the most distant location from the original block. The method significantly reduced the tamper coincidence problem, contributing to enhanced recovered image quality. In addition, the AuSR3 utilized an improved LSB shifting algorithm for watermark embedding, resulting in higher-quality watermarked images with better PSNR and SSIM values than previous techniques.

2.2. Multiple recoveries

Multiple recoveries could be used in conjunction with the block mapping technique to reduce the tamper coincidence. This technique embedded multiple recovery data in different image blocks to ensure that if one recovery data is destroyed, another recovery data is expected to recover the original image block. This technique can significantly reduce the tamper coincidence in any tampering ratio. However, storing multiple recoveries requires more space for each image block, requiring a larger block size to embed all the watermark data. At the same time, larger block sizes may reduce the accuracy and precision of the tamper localization. For example, in a single recovery technique with a block

size of 2 × 2 pixels, if one pixel is tampered with, the remaining three pixels are also marked as tampered, resulting in a 75 % false positive rate. In comparison, the multiple recoveries technique with a block size of 3 × 3 pixels, if one pixel is modified, the remaining 8 pixels are also considered tampered with, producing an 89 % false positive rate. High false positive rates lead to lower precision and accuracy. Several techniques have been developed to embed the recovery data in multiple recoveries.

Haghighi et al. [16] presented a fragile blind quad watermarking scheme for image tamper detection and recovery, utilizing Lifting Wavelet Transform (LWT) and Genetic Algorithm (GA). The scheme generated four compact digests with high quality for tamper recovery, providing multiple chances for restoring damaged blocks. It employed a unique parameter estimation technique based on GA to optimize the quality of digests and the watermarked image. Additionally, it used the Chebyshev System to embed, encrypt, and shuffle information, enhancing security and recovery rates.

Aminuddin and Ernawan [17] presented AuSR2, an image watermarking technique for authentication and self-recovery, ensuring the integrity of digital images against forgery. It embedded watermark data into non-overlapping blocks of an image into 2-LSB, which included authentication and recovery bits. The technique preserved the texture of each block, allowing independent recovery of tampered pixels. Multiple recovery data were embedded, including maximum and minimum pixel values for each block and its texture information, to reduce the tamper coincidence problem. The method demonstrated high accuracy in tamper detection and high quality in recovered images.

Renklier and Öztürk [18] presented a self-embedding fragile watermarking method for image authentication and recovery. This method utilized a Sudoku puzzle and MD5 hash algorithm. The grayscale image was divided into 25 blocks based on a 5 × 5 Sudoku puzzle, and each block was further divided into non-overlapping sub-blocks. The recovery information was derived from the average pixel values of these sub-blocks. The MD5 hash algorithm was used for authentication, considering the block position, pixel values, and a secret key. The tamper coincidence was addressed by embedding four copies of the sub-block recovery information into different blocks based on the Sudoku puzzle. This redundancy ensured that the chances of both a block and its recovery information being tampered with simultaneously were minimized.

Molina-Garcia et al. [19] presented a fragile watermarking scheme for color-image authentication and self-recovery. The scheme used a hierarchical tamper detection algorithm to increase the accuracy of identifying altered regions in the image. The paper also presented a solution to the tamper coincidence, which occurs when the image block and its recovery watermark are tampered with. The scheme embedded three recovery watermarks in different positions to mitigate this issue. It increases the likelihood of reconstructing the original content, even if some parts of the image are tampered with. Finally, to preserve the quality of the image, the scheme employed bilateral filtering and an inpainting algorithm. These techniques help maintain image quality during noise suppression and edge preservation.

2.3. Image inpainting

Unlike the block mapping and multiple recoveries implemented during the watermark embedding stage, the image inpainting technique is implemented during the self-recovery stage. Since it works in different stages of image watermarking, this image inpainting technique could also be used in conjunction with the block mapping and multiple recoveries technique. The image inpainting technique repairs the tampered coincidence blocks using the information from the surrounding blocks. The most straightforward implementation of image inpainting is computing the average value of 8 blocks surrounding the tamper coincidence block. This average value is then employed to replace the block with a tamper coincidence in the middle. However, when the tamper coincidence is concentrated, the neighboring block may also suffer from the tamper coincidence. Thus, the technique should also consider interpolating the further neighboring block until it is available. In such cases, the image inpainting will take a high computational time to interpolate the tamper coincidence block. Several techniques with image inpainting capabilities have been developed to repair the tamper coincidence.

Ernawan et al. [20] presented BRIWT, a blind recovery technique for image watermarking using integer wavelet transforms. The scheme embedded recovery data into the two Least Significant Bits (LSB) of the image content using the LSB adjustment technique. The scheme employed the three-layer authentication to validate the integrity of image contents, which resulted in high precision and accuracy in tamper localization. Furthermore, the research investigated an image inpainting method to enhance recovery from tampered images by identifying non-tampered pixels in the surrounding tamper localization. To solve the tamper coincidence, the scheme used information from surrounding blocks to restore the tampered block.

Xia et al. [21] presented a method for color image tampering detection and self-recovery based on fragile watermarking. This method aimed to enhance the quality of watermarked images, improve tamper detection, and improve the quality of content recovery. The watermark embedding process used block-based regular markers with pixel-based continuous markers to enhance the quality of watermarked images and adapt to various scenarios. The paper introduced a feature extraction-based tampering detection scheme for diagonal blocks with three-layer authentication capable of resisting diverse tampering attacks. It also introduced a block pixel-level recovery mechanism and an improved smoothing inpainting algorithm to recover tampered images. The tampering coincidence problem was addressed by embedding recovery data in other blocks and replacing the tampered blocks during recovery.

Huo et al. [22] presented an alterable-capacity watermarking scheme that could restore images for authentication purposes. This scheme generated a code based on the roughness of image blocks, allowing for varying lengths of watermarks. It included methods for detecting tampering and restoring the image using two copies of significant code and an image inpainting method. The tamper coincidence was addressed by embedding the significant code in different blocks and using image inpainting for blocks with destroyed codes.

Al-Otum and Ellubani [23] presented a dual watermarking technique for color image tampering detection and self-restoration. This technique combined robust watermarking, which is used for copyright protection, and fragile watermarking, which is used for tamper detection and recovery. To enhance security, the watermark bits were distributed across the RGB layers using a secret seed number-based selector, a method referred to as selective channel watermarking. The paper also presented a hierarchical tamper detection algorithm and a modified block-based 2-LSB approach, which ensured high accuracy in detecting and recovering tampered areas. The paper addressed the tamper coincidence by filling these regions with the average value of successfully recovered neighboring pixels. This was done as part of the post-processing techniques, including inpainting and bilateral filtration, to suppress noise and preserve edge content.

3. Proposed method

The image authentication and self-recovery framework consists of three main stages: watermark embedding, tamper localization, and self-recovery. The watermark embedding stages require generating the block map and the watermark data before it is finally embedded into the host image. Once the watermark data is embedded, the watermarked image can be sent securely to the intended recipient. Next, the recipient extracts the watermark data to authenticate the image. If the image has been tampered with, tamper localization is applied to localize the tampered region of the image. Tamper localization can be performed if

the watermark data has been extracted and reconstructed from the tampered image. In addition, the block map should be reconstructed identically as in the watermark embedding stage. The final step is the self-recovery stage, which requires tamper coincidence localization. If a block of the image suffers the tamper coincidence, the block is then interpolated using the image inpainting technique; otherwise, the extracted recovery data can be used for recovery. The diagram and evaluation flow of the image authentication and self-recovery framework are illustrated in Fig. 1.

Image authentication and self-recovery framework are currently evaluated in three performance evaluations: watermarked image quality, tamper localization, and recovered image quality. The image quality is evaluated based on its imperceptibility using PSNR and SSIM. The tamper localization is evaluated using the confusion matrix, which can be utilized further to calculate the precision and accuracy values. This paper proposes new evaluation metrics to compute the tamper coincidence. The proposed evaluation techniques compute the TCBR and TCBD values based on the tamper coincidence localization.

3.1. Tamper coincidence evaluation

The tamper coincidence occurs when the original and corresponding recovery blocks are modified at the same time. This paper proposes two evaluation metrics for these tamper coincidences: Tamper Coincidence Block Ratio (TCBR) and Tamper Coincidence Block Density (TCBD). TCBR and TCBD are defined as follows:

$$TCBR = \frac{TCB}{M \times N} \tag{1}$$

$$TCBD = \frac{1}{TCB} \sum_{i=1}^{TCB} \frac{s_i}{b_i} \tag{2}$$

where TCB represents the number of tamper coincidence blocks in the tampered image, M and N indicate the width and the height of the block map, $TCBR$ represents the ratio of the tamper coincidence block, b_i is the number of surrounding blocks of i, s_i is the number of TCB in bi, and $TCBD$ represents the average density of the tamper coincidence block. The algorithm to compute the TCBR and TCBD is defined in Algorithm 1.

Algorithm 1 TCBR and TCBD Algorithm

Input: Tamper coincidence localization

1	$[M, N] = \text{size}(Input)$
2	$[TCB, \Sigma S_i B_i] = 0$
3	**for** $x = 1$ to M
4	**for** $y = 1$ to N
5	**if** $Input(x, y) ==$ TRUE **then**
6	$TCB{=}TCB{+}1$
7	$x_1 = \max(1, x - 1)$
8	$x_2 = \min(M, x + 1)$
9	$y_1 = \max(1, y - 1)$
10	$y_2 = \min(N, y + 1)$
11	$Local = Input(x_1{:}x_2, y_1{:}y_2)$
12	$[h, w] = \text{size}(Local)$
13	$s_i = \text{count}(Local == \text{TRUE}) - 1$

(continued on next page)

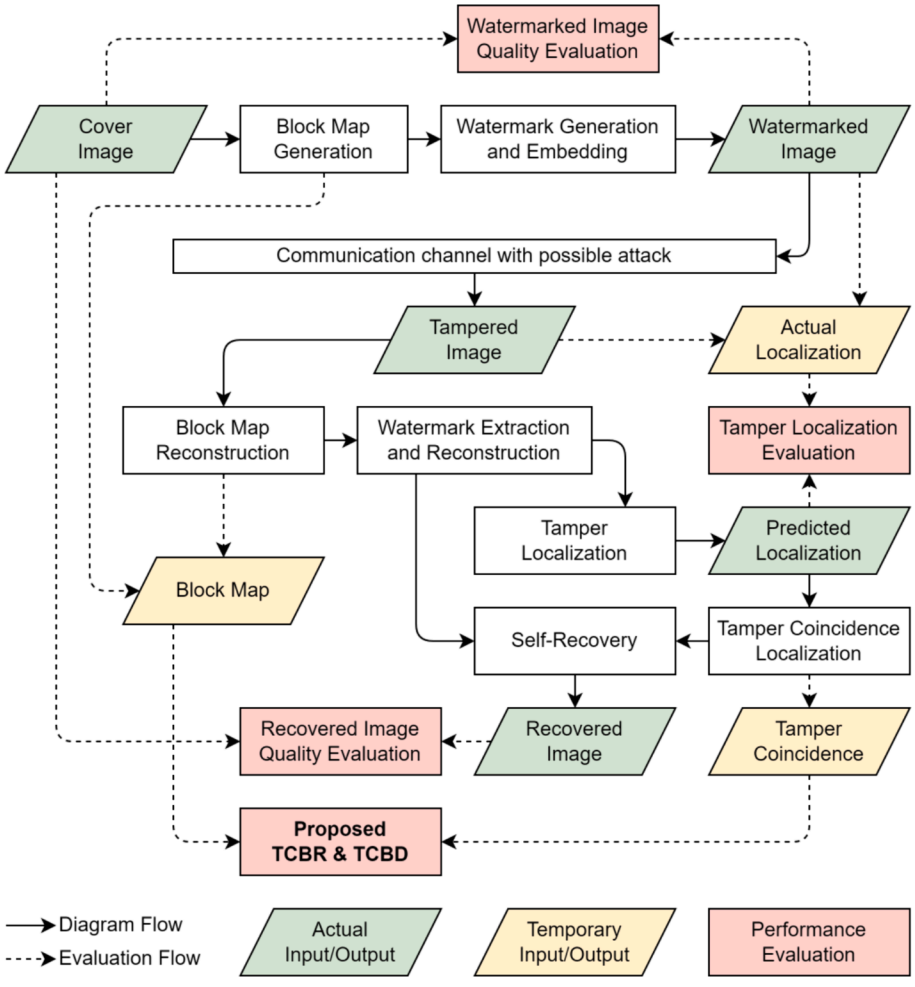

Fig. 1. Image authentication and self-recovery framework with the diagram and evaluation flow.

(*continued*)

Algorithm 1 TCBR and TCBD Algorithm

Input: Tamper coincidence localization

1	$[M, N]$ = size(*Input*)
14	$b_i = (h \times w) - 1$
15	$\Sigma S_i B_i = \Sigma S_i B_i + (s_i / b_i)$
16	**end (if)**
17	**end (for)**
18	**end (for)**
19	$TCBR = TCB / (M \times N)$
20	$TCBD = \Sigma S_i B_i / TCB$

Output: *TCBR*, *TCBD*

The input of Algorithm 1 is a binary matrix with the size of M×N. Each matrix cell contains a binary value of 0 or 1. 0 represents a non-tamper coincidence block, while 1 represents a tamper coincidence block. At first, the input is divided into overlapping blocks with the size of 3 × 3 for a center block, 2 × 2 for a corner block, and 2 × 3 or 3 × 2 for an edge block, as shown in Lines 7 to 11. This overlapping block is also called a local block. The size of the local block is computed in Line 12. The local block still contains the original tamper coincidence block in the current location. Therefore, the values of s_i and b_i are substracted by 1 in Lines 13 and 14. Finally, TCBR and TCBD are computed in Lines 19 and 20, respectively.

The value of TCBR and TCBD is ranging from 0 to 1. The value of 0 represents no tamper coincidence occurred, while 1 represents a complete tamper coincidence. In this case, recovering the image with a complete tamper coincidence is impossible, as all recovery data are corrupted. Therefore, the TCBR and TCBD values can provide insight to the researcher when evaluating their block map. Furthermore, the researcher could design an effective block mapping technique to prevent tamper coincidence and enhance the recovered image quality.

3.2. Tamper coincidence simulation

The image with the tamper coincidence is simulated in Fig. 2. It divides the image into 64 blocks. Each block is sequentially numbered (written in black) between 1 and 64 to show the original block location. In addition, the block map also decides the recovery data location of each block (written in blue). For example, the number 49 is written in block number 1. This means that the recovery data of block 49 will be embedded into the LSB of block 1. At the same time, the recovery data of block 1 will be embedded into the LSB of block 25. The block map depicted in Fig. 2 uses random block map techniques as presented in [14] and [20].

Fig. 2 shows that eight blocks suffer from the tamper coincidence (blocks 1, 5, 9, 12, 17, 20, 21, and 26). This is because the recovery data of these blocks are stored inside the tampered region of the image. For example, block 1 stores the recovery data into block 25, while block 5 stores the recovery data into block 33. In this case, both blocks (the original block location and the recovery data location) are tampered with at the same time. According to the simulated random block map in Fig. 2, the block map has eight tamper coincidence blocks from 64 blocks corresponding to 0.125 TCBR value. Furthermore, the TCBD value of Fig. 2 is computed in Table 1.

The b_i values in Table 1 are computed depending on the block location. If a block is located in the corner of the image, such as blocks 1, 8, 57, and 64, the block has a b_i value of 3. It represents the number of the closest block to the corner block. If a block is situated on the edge of the image, such as block 2 to block 7, the block has a b_i value of 5. The remaining block, which is located in the center of the image, has a b_i value of 8. According to Table 1, the final TCBD value of the simulated random block map is 0.2917.

Table 1
TCBD value of a random block map.

Block	Location	b_i	TCB in b_i	s_i	s_i / b_i
1	Corner	3	9	1	0.3333
5	Edge	5	12	1	0.2000
9	Edge	5	1, 17	2	0.4000
12	Center	8	5, 20, 21	3	0.3750
17	Edge	5	9, 26	2	0.4000
20	Center	8	12, 21	2	0.2500
21	Center	8	12, 20	2	0.2500
26	Center	8	17	1	0.1250
Average					0.2917

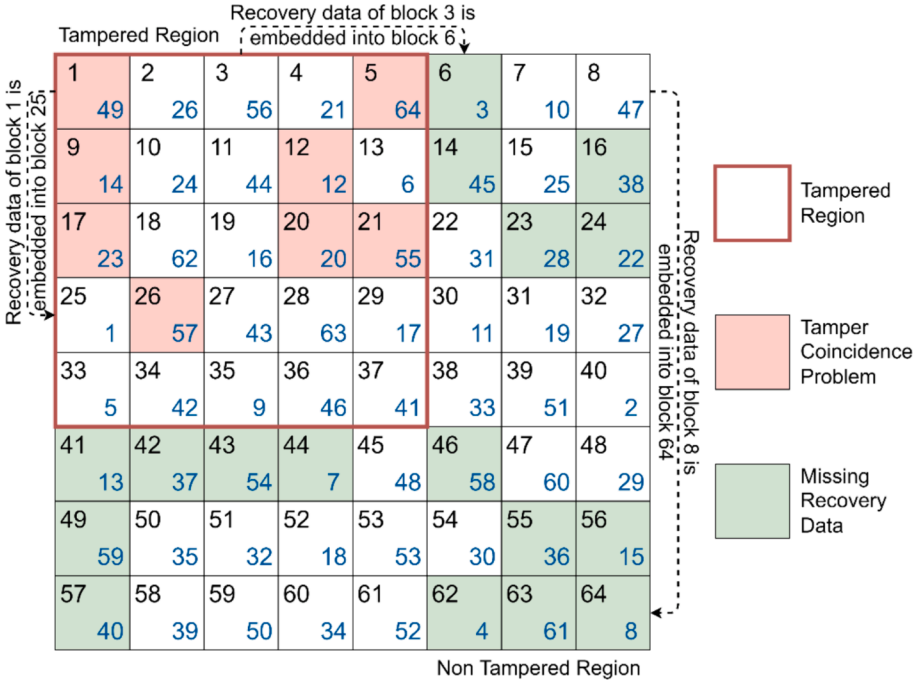

Fig. 2. Simulated random block map with the tamper coincidence.

3.3. Tamper localization evaluation

In image tamper localization, a confusion matrix can be used to evaluate the performance of algorithms designed to detect and localize tampered regions within an image. The confusion matrix helps quantify the accuracy of the algorithm in identifying true tampered regions and distinguishing them from non-tampered regions. The confusion matrix typically compares the predicted tampered regions against the actual (ground truth) tampered regions. For a binary classification problem (tampered vs. non-tampered pixels), the confusion matrix can be represented as True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). From the confusion matrix, precision and accuracy can be derived to evaluate the tamper localization:

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

where *TP* represents the number of pixels correctly identified as tampered, *TN* denotes the number of pixels correctly identified as non-tampered, *FP* signifies the number of pixels incorrectly identified as tampered (false alarms), and *FN* shows the number of pixels incorrectly identified as non-tampered (missed tampered pixels).

3.4. Image quality evaluation

In the image authentication and self-recovery framework, the quality of the recovered image can be computed using PSNR and SSIM. PSNR measures the quality of the recovered image by computing the ratio of the maximum possible power of the signal to the power of corrupting noise that affects the fidelity of its representation. In comparison, SSIM measures the perceived changes in structural, contrast, and luminance information between images. High PSNR and SSIM values indicate a high recovered image quality [24]. The PSNR can be computed as follows [25]:

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX^2}{MSE}\right) \tag{5}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(O(i) - R(i))^2 \tag{6}$$

where *MAX* represents the maximum possible pixel value of the image (255 for an 8-bit image), *MSE* (Mean Squared Error) is the average squared difference between the cover and recovered images, *n* denotes the total number of pixels in the image, $O(i)$ and $R(i)$ signifies the intensity of the i-th pixel in the cover and recovered image, respectively. The SSIM can be computed as follows [25]:

$$SSIM = \frac{2\mu_O\mu_R + C_1}{\mu_O^2 + \mu_R^2 + C_1} \cdot \frac{2\sigma_O\sigma_R + C_2}{\sigma_O^2 + \sigma_R^2 + C_2} \cdot \frac{\sigma_{OR} + C_3}{\sigma_O\sigma_R + C_3} \tag{7}$$

where *O* and *R* are two images being compared, μ represents the pixel sample mean of the image, σ^2 denotes the variance of the image, σ signifies the covariance of the image, and *C1*, *C2*, and *C3* are constants to stabilize the division.

4. Experimental results

The experiments carried out in this research are performed using the USC-SIPI dataset with eight color images, each with a size of 512 × 512 pixels, as shown in Fig. 3. The experiment measures the TCBR and TCBD of four existing image authentication and self-recovery techniques [14,15,17,20]. At first, all eight images in the dataset undergo watermark embedding stages to produce the watermarked images.

The watermarked images are then tampered with using a regular attack by adding noise in the central region of the images. The noise ranges between 0 to 100 % with the step of 10 %. The noise is consistently generated using the same seed value to ensure that all images and watermarking techniques utilize the same experimental setup. The tampered images with regular attacks are shown in Fig. 4.

In addition to the regular attack, irregular attacks are applied to the watermarked images to provide insight into how the technique performs in a real-life scenario. The irregular attacks include copy-move forgery, image splicing, content removal, and text addition. Copy-move forgery involves copying a portion of an image and pasting it into another location within the same image. This type of forgery is often used to duplicate or obscure content. Image splicing involves combining elements from two or more images to create a single composite image. This technique is commonly used in digital forgeries to insert objects or people into a scene. Content removal involves deleting specific content from an image and filling in the resulting gaps to make the image appear natural. This forgery often removes unwanted objects or people from an image. Text addition involves inserting text into an image, which can be

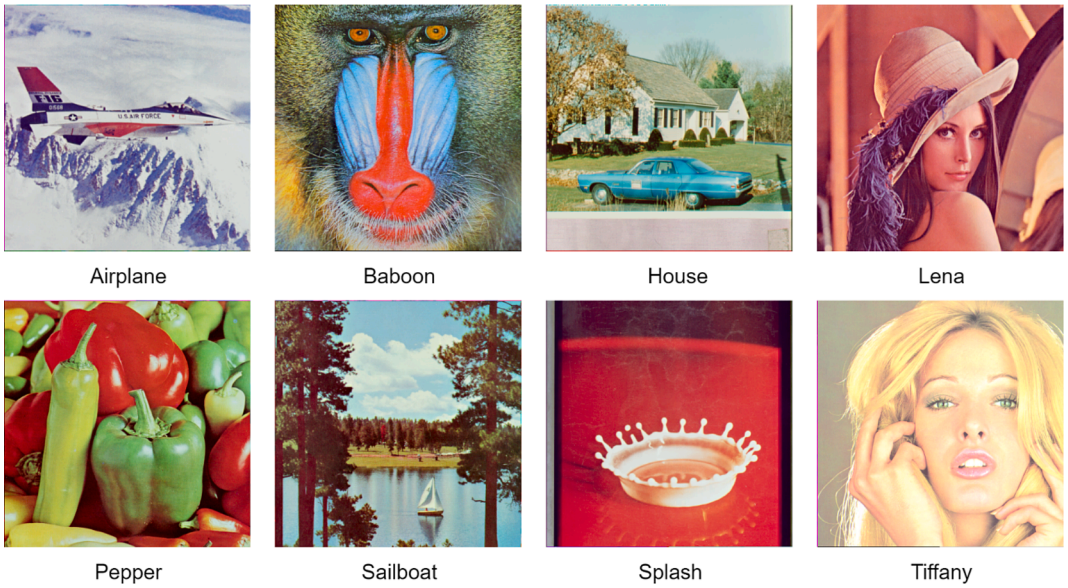

Fig. 3. USC-SIPI color image datasets.

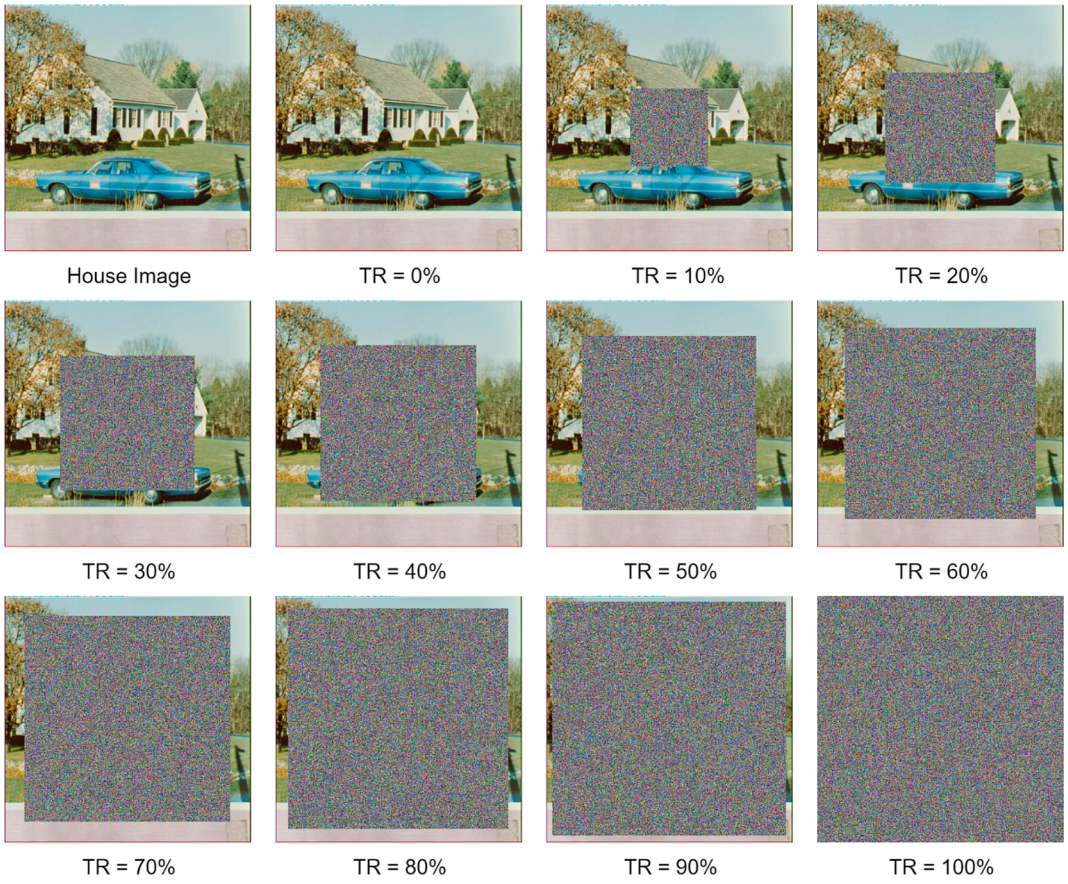

Fig. 4. The tampered house image with the tampering rate (TR) between 0% and 100%.

used for various purposes, such as adding captions, labels, or false information. The tampered images with irregular attacks are shown in Fig. 5.

4.1. Tamper localization

The tampered image then undergoes a tamper localization process, which localizes the tampered region of the image. The performance of the tamper localization is measured using the confusion matrix, providing values to compute precision and accuracy. A potential issue that may affect the precision and accuracy of tamper localization is that the confusion matrix is computed on a pixel basis despite the watermark embedding technique being performed on a block basis. Therefore, if one pixel on the image block is modified, the remaining pixels will be considered false positive detection. Another issue in tamper localization is that the scheme has limited space for embedding the authentication data for each block, as the recovery data will occupy the most space for the self-recovery process. If an image block only stores one bit of authentication data, the possibility of true positive detection is 50 %. The possibility increases to 75 % when two bits of authentication data are embedded and 87.5 % when three bits are used. Essentially, more authentication data embedded will provide high true positive detection. However, due to the limited space for embedding, the researcher limited the number of authentication data to 2 bits per block [14]. Various techniques have been developed to increase the true positive detection of tamper localization, such as three-layer authentication [17] and hierarchical tamper detection [19]. A higher true positive detection will eventually increase the precision and accuracy of tamper localization. The precision and accuracy of the tamper detection are shown in Tables 2 and 3.

According to Tables 2 and 3, the precision and accuracy do not correlate to the tampering rate (TR) of the image. It can be noticed that most of the techniques presented here have precision and accuracy close to 1 despite the tampering rates ranging between 10 % and 90 %. For example, the AuSR1 obtained the precision and accuracy 1 in 10 %, 50 %, 80 %, and 90 % tampering rates. This is because precision and accuracy are significantly affected by the shape of the tampering area that crosses the image block. If the tampering area matches the block division, the precision and accuracy will be close to 1. On the contrary, the precision and accuracy will be lower if the tampering area is cut through the middle of the block, as it may produce false positive tamper detection. Therefore, designing efficient tamper localization techniques is required to handle false negative and false positive detection, increasing precision and accuracy.

4.2. Tamper coincidence problem

The tampered image then undergoes a tamper coincidence localization, which localizes the tamper coincidence of the image. The tamper coincidence localization is then measured using the proposed TCBR and TCBD. The tamper localization and tamper coincidence localization are shown in Fig. 6. It can be seen that all four techniques can precisely localize the tampered region of the images with high precision and accuracy. However, the ability to prevent the tamper coincidence differs between those four techniques. In addition, each RGB channel has a different tamper coincidence localization caused by different seed values used when generating the block map. The localization of the tamper coincidences shown in the white spots in Fig. 6 is obtained from the blue channel of the image.

The AuSR1 [14] and BRIWT [20] only implemented a random block mapping technique to prevent tamper coincidence. Therefore, this technique suffers the most severe tamper coincidence. In contrast,

Fig. 5. Irregular attacks in the USC-SIPI dataset with the tamper localization (TL), the tamper coincidence localization (TCL), and the recovered image (RI).

Table 2
Precision comparison of the tampered images.

TR	Fan [9]	Tai [8]	Molina-Garcia [19]	Reyes-Reyes [26]	AuSR1 [14]	AuSR2 [17]	AuSR3 [15]	BRIWT [20]
10	0.8007	0.9670	0.9152	0.9157	1.0000	0.9986	1.0000	1.0000
20	0.9210	0.9855	0.9580	0.9585	0.9978	0.9934	0.9978	0.9978
30	0.9144	0.9903	0.9716	0.9718	0.9939	0.9909	0.9939	0.9939
40	0.9483	0.9939	0.9797	0.9799	0.9918	0.9959	0.9918	0.9918
50	1.0000	1.0000	0.9884	0.9885	1.0000	0.9890	1.0000	1.0000
60	0.9601	0.9943	0.9848	0.9849	0.9925	0.9925	0.9925	0.9925
70	0.9748	0.9958	0.9876	0.9877	0.9787	0.9785	0.9787	0.9787
80	0.9659	0.9963	0.9891	0.9892	1.0000	0.9500	1.0000	1.0000
90	0.9762	0.9972	0.9909	0.9910	1.0000	0.9302	1.0000	1.0000

AuSR3 [15] implemented an advanced block mapping technique to ensure the recovery data is embedded into the furthest location. Using this technique, the tamper coincidence will only appear when the tampering rate exceeds 25 %. The AuSR2 [17] implemented multiple recoveries to prevent the tamper coincidence. Thus, when one recovery data suffers the tamper coincidence, another will recover the missing recovery data. Despite implementing multiple recoveries, the tamper coincidence may still manifest if all the recovery data are compromised.

Table 3
Accuracy comparison of the tampered images.

TR	AuSR1 [14]	AuSR2 [17]	AuSR3 [15]	BRIWT [20]
10	1.0000	0.9993	1.0000	1.0000
20	0.9989	0.9967	0.9989	0.9989
30	0.9969	0.9954	0.9969	0.9969
40	0.9959	0.9979	0.9959	0.9959
50	1.0000	0.9944	1.0000	1.0000
60	0.9962	0.9962	0.9962	0.9962
70	0.9891	0.9890	0.9891	0.9891
80	1.0000	0.9737	1.0000	1.0000
90	1.0000	0.9625	1.0000	1.0000

The tamper coincidence block ratio (TCBR) of the existing techniques is summarized in Table 4.

Table 4 shows that on a 0 % tampering rate, the TCBR value is 0, which represents no tamper coincidence occurred. While on a 100 % tampering rate, the TCBR value is 1, which signifies that all of the recovery data is corrupted, making it impossible to recover the tampered region of the image. The AuSR3 [15] outperforms other techniques regarding the TCBR values due to their advanced block mapping technique. Moreover, on a 10 % and 20 % tampering rate, the TCBR value can be reduced to zero compared to random block mapping techniques such as AuSR1 [14] and BRIWT [20]. The technique that embeds multiple recoveries can also decrease the TCBR value, such as AuSR2 [17]. However, it is insignificant compared to a well-designed block mapping technique of the AuSR3 [15]. In addition, multiple recoveries may take more space for embedding, which enforces larger block sizes, leading to a high false positive detection and lower recovered image quality. In terms of TCBD values, the AuSR2 [17] outperforms other techniques in most scenarios. This is because the redundancy can reduce the density of the tamper coincidence. In contrast, the AuSR3 [15] compresses the tamper coincidence into the edges of the tampered region, increasing the overall density of the tamper coincidence. In comparison, the AuSR1 [14] and BRIWT [20] scattered the tamper coincidence all over the tampered region of the image. The TCBD values of the existing techniques are shown in Table 5.

The image authentication and self-recovery framework implements a post-processing technique to recover the tamper coincidence using the image inpainting technique. This technique is required as the tamper coincidence can not be eliminated entirely when the tampering rate is more significant than 25 %. This technique will collect several neighboring blocks that do not suffer the tamper coincidence to interpolate the recovery value. As a result, a dense tamper coincidence takes more computational power for recovery than a less dense one. In addition, when the tamper coincidence is concentrated in a large region, the interpolated recovery data may not be as precise as the original recovery data. Therefore, the block map should be designed to balance the TCBR and TCBD values to provide a precise recovery while maintaining a low computational power for recovery.

4.3. Recovered image quality

The recovered image quality is measured using PSNR and SSIM. High PSNR and SSIM values signify a high-quality recovered image. In the self-recovery process, the tamper coincidence should be localized as it may significantly affect the recovered image quality. The tamper coincidence localization and the recovered image of the existing techniques are shown in Fig. 7. The baboon image in Fig. 7 is tampered with a 70 % tampering rate. The eye and down of the baboon are inside the tampered

Table 4
TCBR values comparison of the tampered images.

TR	AuSR1 [14]	AuSR2 [17]	AuSR3 [15]	BRIWT [20]
0	0.0000	0.0000	0.0000	0.0000
10	0.0193	0.0134	0.0000	0.0195
20	0.0741	0.0537	0.0000	0.0732
30	0.1572	0.1195	0.0569	0.1570
40	0.2621	0.2085	0.1814	0.2631
50	0.3745	0.3284	0.3088	0.3757
60	0.5089	0.4582	0.4571	0.5104
70	0.6432	0.6129	0.6011	0.6443
80	0.7679	0.7631	0.7359	0.7687
90	0.8921	0.8952	0.8739	0.8922
100	1.0000	1.0000	1.0000	1.0000

Table 5
TCBD values comparison of the tampered images.

TR	AuSR1 [14]	AuSR2 [17]	AuSR3 [15]	BRIWT [20]
0	0.0000	0.0000	0.0000	0.0000
10	0.1896	0.1768	0.0000	0.1942
20	0.3598	0.3393	0.0000	0.3558
30	0.5122	0.4614	0.5367	0.5104
40	0.6414	0.5685	0.6700	0.6435
50	0.7434	0.6756	0.7599	0.7461
60	0.8362	0.7697	0.8372	0.8385
70	0.9055	0.8634	0.8939	0.9071
80	0.9534	0.9340	0.9370	0.9542
90	0.9840	0.9768	0.9712	0.9841
100	1.0000	1.0000	1.0000	1.0000

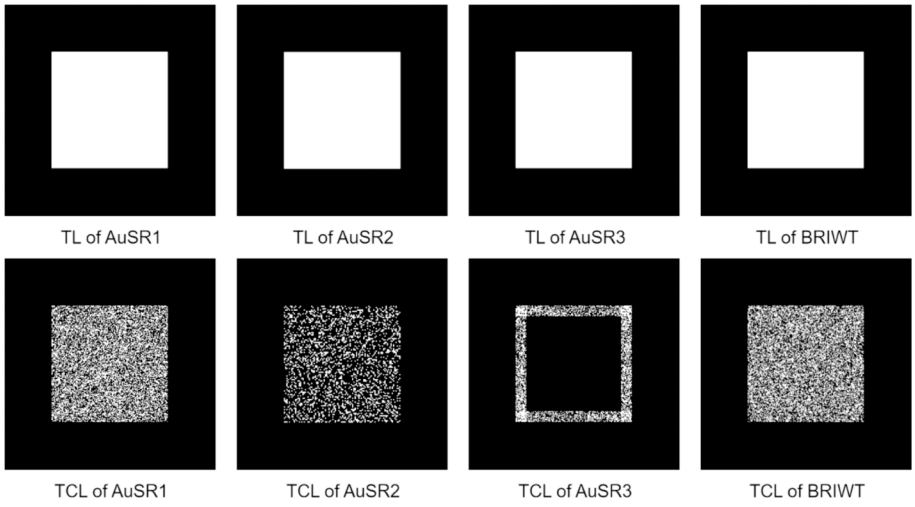

Fig. 6. Tamper localization (TL) and tamper coincidence localization (TCL) with a tampering rate of 30%.

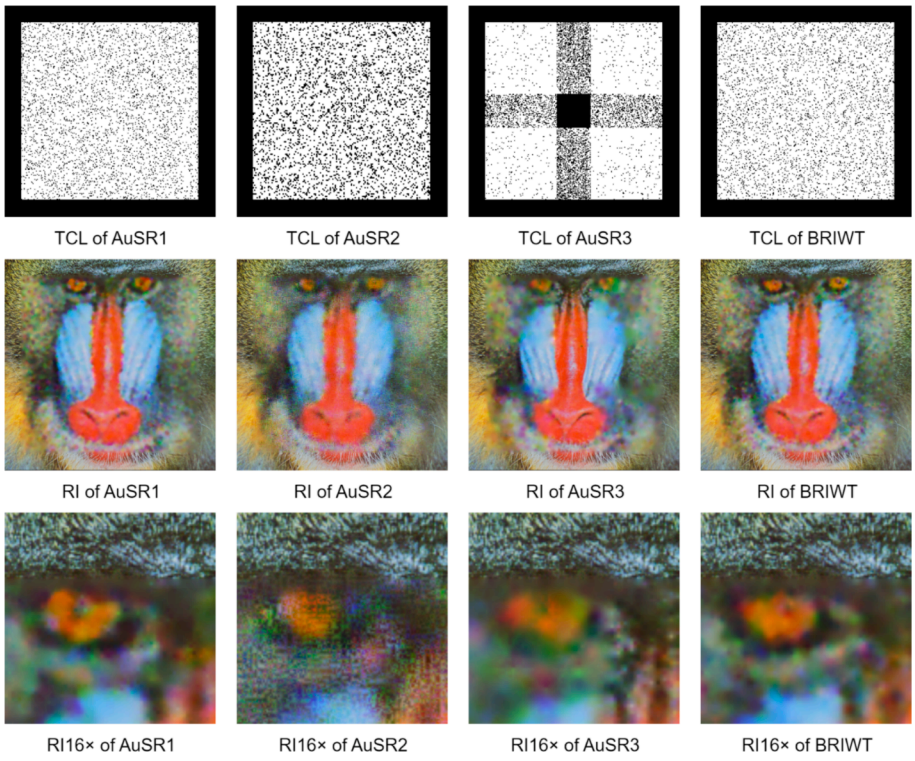

Fig. 7. Tamper coincidence localization (TCL) and the recovered image (RI) with a tampering rate of 70%.

region, which is recovered by the existing techniques. It can be seen that the recovered image quality of the AuSR1 [14] is quite similar to the AuSR3 [15]. This is because both methods use a 2 × 2 block size in which the recovery value is obtained from the average value of the block. Thus, blurred artifacts are shown in the recovered image. The BRIWT [20] also divided the image into non-overlapping blocks with the size of 2 × 2 pixels. The recovery data is obtained from the most significant IWT coefficients to preserve the information of the image in wavelet form. Blurred artifacts are found in the recovered image due to the limited embedding capacity to store all the wavelet information. The overall quality of the recovered image is shown in Tables 6 and 7.

The recovered image quality of the AuSR2 [17] has more details since it uses texture preservation techniques with block sizes of 3 × 3 pixels. Theoretically, the texture preservation technique should produce a higher recovered image quality than the average recovery technique since this technique also stores the texture information of each image block. However, with a higher tampering rate, the overall recovered image quality of the texture preservation technique is lower than the average recovery technique. This is because the tamper coincidence has corrupted the essential information, including the texture information of each block.

The scheme by Fan [9] divided the image blocks and compressed them using the SPIHT algorithm. The output bits are used for self-recovery. In addition, the scheme also generated two versions of recovery data, allowing for reciprocal error correction. If one version is compromised due to tampering, the other can be used to recover the original content. The scheme by Tai [8] implemented a two-level self-recovery scheme for restoring tampered image regions. In level-1 self-recovery, invalid blocks are restored using embedded recovery data and Haar wavelet coefficients. Level-2 self-recovery considers neighboring blocks to recover any remaining invalid ones when required. The scheme by Molina-Garcia [19] embedded three recovery watermarks to address the tampering coincidence problem. The tampered image can still be recovered if one copy of the recovery watermarks survives. In addition, image inpainting and bilateral filtering are employed in the post-processing stage. The scheme by Sinhal [5] applied two smoothing operations to improve the visual quality of the restored image. Despite the smooth recovered image, it also introduces blur artifacts into the recovered image.

Table 6
PSNR values comparison of the recovered images.

TR	Fan [9]	Tai [8]	Molina-Garcia [19]	Sinhal [5]	AuSR1 [14]	AuSR2 [17]	AuSR3 [15]	BRIWT [20]
0	N/A	N/A	N/A	N/A	45.57	45.91	46.20	46.20
10	31.47	25.89	37.34	36.18	37.96	38.11	39.11	38.05
20	28.36	20.57	33.98	32.39	34.65	34.21	36.93	34.76
30	21.62	17.43	31.28	29.99	31.79	31.10	33.46	31.88
40	15.79	15.21	28.47	28.34	29.48	28.63	30.06	29.55
50	15.69	13.54	26.00	27.02	27.64	26.43	27.61	27.68
60	11.57	12.01	23.51	25.46	25.72	24.60	25.16	25.83
70	11.57	10.80	21.23	23.95	23.80	22.66	22.94	23.97
80	8.10	9.81	19.20	22.47	21.63	20.64	20.71	21.92
90	N/A	N/A	N/A	N/A	18.35	18.10	17.86	19.16
100	N/A	N/A	N/A	N/A	4.45	4.45	4.45	4.45

Table 7
SSIM values comparison of the recovered images.

TR	Fan [9]	Tai [8]	Molina-Garcia [19]	Sinhal [5]	AuSR1 [14]	AuSR2 [17]	AuSR3 [15]	BRIWT [20]
0	N/A	N/A	N/A	N/A	0.9972	0.9975	0.9978	0.9978
10	0.9731	0.9384	0.9714	0.9878	0.9928	0.9935	0.9944	0.9934
20	0.9502	0.8443	0.9390	0.9768	0.9864	0.9864	0.9913	0.9872
30	0.8875	0.7364	0.8977	0.9638	0.9742	0.9734	0.9811	0.9751
40	0.7230	0.6226	0.8368	0.9504	0.9555	0.9534	0.9601	0.9567
50	0.7202	0.5135	0.7571	0.9358	0.9339	0.9255	0.9360	0.9355
60	0.4249	0.3899	0.6460	0.9128	0.9059	0.8932	0.9038	0.9078
70	0.4249	0.2744	0.5157	0.8843	0.8705	0.8490	0.8635	0.8737
80	0.0094	0.1655	0.3958	0.8528	0.8219	0.7937	0.8120	0.8280
90	N/A	N/A	N/A	N/A	0.7324	0.7080	0.7220	0.7516
100	N/A	N/A	N/A	N/A	0.0001	0.0001	0.0001	0.0001

4.4. Metrics correlation

The correlation between the metrics is analyzed to gain insight into how the proposed evaluation metrics compare to the existing metrics. The metrics analyzed here are only those that have a positive or negative correlation to the overall tampering rates. If the metrics do not correlate to the tampering rates, it will not provide insightful knowledge to develop a better technique for producing a high-quality recovered image. For example, the precision and accuracy of tamper localization do not correlate to the overall tampering rates. Therefore, the correlation analysis will not provide insight into the recovered image quality. The metrics that have a positive or negative correlation to the overall tampering rates are PSNR, SSIM, TCBR, and TCBD. The correlations between PSNR, SSIM, TCBR, and TCBD are presented in Fig. 8.

Fig. 8 shows that the PSNR and SSIM have a positive correlation. This means that a high PSNR value corresponds to a high SSIM value. The TCBR and TCBD also have a positive correlation. Thus, the increase of TCBR value also increases the TCBD values. In comparison, the correlation between PSNR and TCBR or TCBD is negative. This means that when TCBR and TCBD values increase, the PSNR value decreases, which signifies a lower recovered image quality. It is possible since the increase of TCBR and TCBD signifies a significant amount of recovery data are corrupted, leading to a lower recovered image quality. The SSIM also negatively correlates to the TCBR and TCBD. The higher the TCBR and TCBD values, the lower the SSIM values. This correlation indicates that the tamper coincidence decreases the recovered image quality.

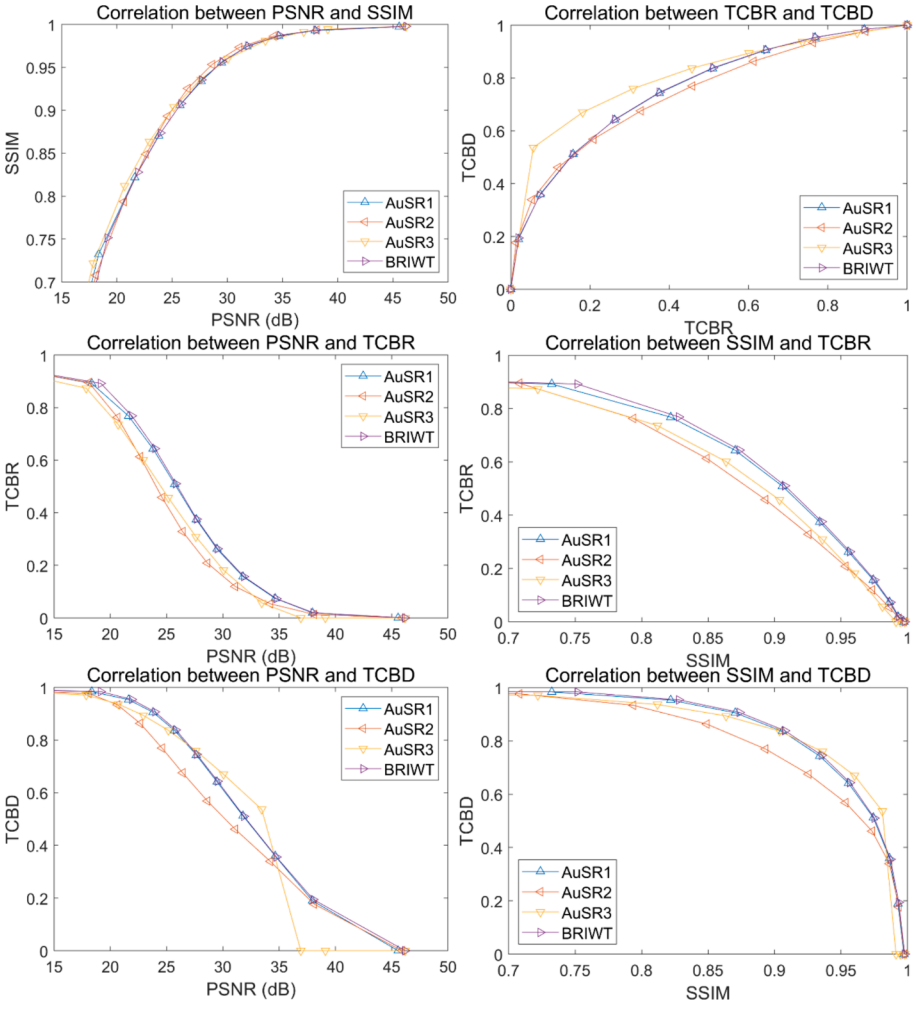

Fig. 8. The correlations between PSNR, SSIM, TCBR, and TCBD values.

Therefore, the researcher should design a block mapping technique to minimize the tamper coincidence while increasing the recovered image quality. In addition, the TCBR and TCBD values also provide an insightful evaluation when designing an effective block mapping technique for embedding the recovery data.

5. Conclusion

This paper has introduced new evaluation metrics of Tamper Coincidence Block Ratio (TCBR) and Tamper Coincidence Block Density (TCBD) to assess the effectiveness of block map designs in preventing tamper coincidence. This research emphasizes the importance of designing an effective block mapping technique to minimize the TCBR and TCBD values. This study also demonstrated that TCBR and TCBD values are negatively correlated to the PSNR and SSIM values of the recovered image. A high TCBR and TCBD value leads to low recovered image quality. This is because a high tamper coincidence limits the capability of the image inpainting technique to recover the tampered region. This paper suggests future research efforts to prioritize refining and optimizing block mapping techniques and mitigate the impact of the tamper coincidence, enhancing the overall effectiveness of image authentication and self-recovery framework.

CRediT authorship contribution statement

Afrig Aminuddin: Writing – original draft, Methodology, Investigation, Conceptualization. **Ferda Ernawan:** Writing – review & editing, Validation, Supervision, Investigation. **Danakorn Nincarean:** Validation, Supervision. **Agit Amrullah:** Writing – review & editing, Validation, Conceptualization. **Dhani Ariatmanto:** Writing – review & editing, Validation.

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.

Acknowledgments

This work was supported by the Ministry of Higher Education for providing financial support under Fundamental Research Grant Scheme (FRGS), No. FRGS/1/2023/ICT04/UMP/02/1 (University reference RDU230121).

References

[1] A.K. Sahu, A logistic map based blind and fragile watermarking for tamper detection and localization in images, J. Ambient Intell. Humaniz. Comput. (Jun. 2021) 1–13, https://doi.org/10.1007/S12652-021-03365-9.

[2] S. Sharma, J.J. Zou, G. Fang, A novel multipurpose watermarking scheme capable of protecting and authenticating images with tamper detection and localisation abilities, IEEE Access 10 (2022) 85677–85700, https://doi.org/10.1109/ACCESS.2022.3198963.

[3] F. Tohidi, M. Paul, M.R. Hooshmandasl, Detection and recovery of higher tampered images using novel feature and compression strategy, IEEE Access 9 (Apr. 2021) 1, https://doi.org/10.1109/access.2021.3072314.

[4] 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) (Nov. 2014) 1197–1210, https://doi.org/10.1016/J.IMAGE.2014.09.001.

[5] R. Sinhal, I.A. Ansari, C.W. Ahn, Blind image watermarking for localization and restoration of color images, IEEE Access 8 (2020) 200157–200169, https://doi.org/10.1109/ACCESS.2020.3035428.

[6] B. Bolourian Haghighi, A.H. Taherinia, A. Harati, TRLH: fragile and blind dual watermarking for image tamper detection and self-recovery based on lifting wavelet transform and halftoning technique, J. Vis. Commun. Image Represent. 50 (Jan. 2018) 49–64, https://doi.org/10.1016/J.JVCIR.2017.09.017.

[7] 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) (Mar. 2013) 301–308, https://doi.org/10.1016/j.image.2012.12.003.

[8] W.L. Tai, Z.J. Liao, Image self-recovery with watermark self-embedding, Signal Process. Image Commun. 65 (Jul. 2018) 11–25, https://doi.org/10.1016/J.IMAGE.2018.03.011.

[9] M.Q. Fan, H.X. Wang, An enhanced fragile watermarking scheme to digital image protection and self-recovery, Signal Process. Image Commun. 66 (Aug. 2018) 19–29, https://doi.org/10.1016/J.IMAGE.2018.04.003.

[10] Z. Shao, Y. Shang, Y. Zhang, X. Liu, G. Guo, Robust watermarking using orthogonal Fourier-Mellin moments and chaotic map for double images, Signal Process. 120 (Mar. 2016) 522–531, https://doi.org/10.1016/J.SIGPRO.2015.10.005.

[11] Y. Chen, Z. Jia, Y. Peng, Y. Peng, Efficient robust watermarking based on structure-preserving quaternion singular value decomposition, IEEE Trans. Image Process. 32 (2023) 3964–3979, https://doi.org/10.1109/TIP.2023.3293773.

[12] Z. Shao, Y. Shang, R. Zeng, H. Shu, G. Coatrieux, J. Wu, Robust watermarking scheme for color image based on quaternion-type moment invariants and visual cryptography, Signal Process. Image Commun. 48 (Oct. 2016) 12–21, https://doi.org/10.1016/J.IMAGE.2016.09.001.

[13] 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 (Jul. 2016) 775–789, https://doi.org/10.1016/j.jvcir.2016.04.023.

[14] 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. 34 (8) (Sep. 2022) 5822–5840, https://doi.org/10.1016/j.jksuci.2022.02.009.

[15] A. Aminuddin, F. Ernawan, AuSR3: A new block mapping technique for image authentication and self-recovery to avoid the tamper coincidence problem, J. King Saud Univ. – Comput. Inf. Sci. 35 (9) (Oct. 2023), https://doi.org/10.1016/j.jksuci.2023.101755.

[16] 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) 486 (Jun. 2019) 204–230, https://doi.org/10.1016/j.ins.2019.02.055.

[17] A. Aminuddin, F. Ernawan, AuSR2: Image watermarking technique for authentication and self-recovery with image texture preservation, Comput. Electr. Eng. 102 (108207) (Sep. 2022) 108207, https://doi.org/10.1016/j.compeleceng.2022.108207.

[18] A. Renklier, S. Öztürk, Image authentication and recovery: Sudoku puzzle and MD5 hash algorithm based self-embedding fragile image watermarking method, Multimed. Tools Appl. 83 (5) (Feb. 2024) 13929–13951, https://doi.org/10.1007/S11042-023-15999-2/TABLES/7.

[19] 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 (Feb. 2020) 115725, https://doi.org/10.1016/j.image.2019.115725.

[20] F. Ernawan, A. Aminuddin, S. Abu Bakar, A blind recovery technique with integer wavelet transforms in image watermarking, Eng. Sci. Technol. Int. J. 48 (101586) (Dec. 2023) 101586, https://doi.org/10.1016/j.jestch.2023.101586.

[21] X. Xia, S. Zhang, K. Wang, T. Gao, A novel color image tampering detection and self-recovery based on fragile watermarking, J. Inf. Secur. Appl. 78 (Nov. 2023) 103619, https://doi.org/10.1016/J.JISA.2023.103619.

[22] Y. Huo, H. He, F. Chen, Alterable-capacity fragile watermarking scheme with restoration capability, Opt. Commun. 285 (7) (Apr. 2012) 1759–1766, https://doi.org/10.1016/J.OPTCOM.2011.12.044.

[23] H.M. Al-Otum, A.A.A. Ellubani, Secure and effective color image tampering detection and self restoration using a dual watermarking approach☆, Optik (Stuttg) 262 (Dec. 2022) 169280, https://doi.org/10.1016/J.IJLEO.2022.169280.

[24] M. Begum, M.S. Uddin, Digital image watermarking techniques: a review, Inf. 11 (2) (Feb. 2020) 110, https://doi.org/10.3390/INFO11020110.

[25] D.R.I.M. Setiadi, PSNR vs SSIM: imperceptibility quality assessment for image steganography, Multimed. Tools Appl. 80 (6) (Mar. 2021) 8423–8444, https://doi.org/10.1007/s11042-020-10035-z.

[26] R. Reyes-Reyes, C. Cruz-Ramos, V. Ponomaryov, B.P. Garcia-Salgado, J. Molina-Garcia, Color image self-recovery and tampering detection scheme based on fragile watermarking with high recovery capability, Appl. Sci. 11 (7) (Apr. 2021) 3187, https://doi.org/10.3390/APP11073187.