Journal of King Saud University – Computer and Information Sciences 35 (2023) 101755

HOSTED BY

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

AuSR3: A new block mapping technique for image authentication and self-recovery to avoid the tamper coincidence problem

Afrig Aminuddin [a,b,*], Ferda Ernawan [a]

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

ARTICLE INFO

Article history:
Received 6 June 2023
Revised 25 August 2023
Accepted 11 September 2023
Available online 15 September 2023

Keywords:
Block mapping
Image authentication
Tamper coincidence problem
Image inpainting
Self-recovery

ABSTRACT

This paper proposes a new block mapping technique for image authentication and self-recovery designed to avoid the tamper coincidence problem called the AuSR3. The tamper coincidence problem can arise when modifications to an image affect the original block and its recovery data, resulting in the inability to recover the tampered region of the image. The new block mapping technique ensures that the recovery data of a block is embedded into the most distant location possible, minimizing the tamper coincidence problem. In addition, the improved LSB shifting algorithm is employed to embed the watermark data consisting of authentication and recovery data. The experimental result shows that the AuSR3 can produce high-quality watermarked images across various datasets with average PSNR values of 46.2 dB, which improved by 2.1 dB compared to the LSB replacement technique. The new block mapping technique avoids the tamper coincidence problem by up to 25% tampering rates. It contributes to the high-quality recovered image with a PSNR and SSIM value of 39.10 dB and 0.9944, respectively, on a 10% tampering rate on the USC-SIPI dataset.

1. Introduction

The development of multimedia technology has revolutionized the way we use and interact with images. One of the technologies is image manipulation software, which refers to various tools and techniques used to alter or enhance digital images. With advancements in computer technology, image manipulation software has grown significantly, expanding its capabilities and finding broader applications across various fields. Nowadays, this software is employed in diverse settings, from basic photo editing to creating highly realistic and intricate visual effects. However, throughout its evolution, image manipulation software has encountered criticism and controversy due to its capacity to deceive and propagate false information. In response to these concerns, the researchers have developed techniques to detect and prevent the dissemination of manipulated images, such as the image authentication technique.

The image authentication technique is a technique for verifying the authenticity of digital images. There are two main categories of image authentication techniques: active and passive. Active image authentication techniques require preliminary data for authentication (Prasad and Pal, 2020; Bolourian Haghighi et al., 2019; Bolourian Haghighi et al., 2018). The preliminary data are commonly represented as the watermark data, which will be embedded into the cover image. This watermark data is imperceptible to the human eye but can be detected using an image authentication algorithm. Later, the watermark data can be extracted and used to localize the tampered region of the image. In contrast, passive image authentication techniques do not involve embedding watermark data. Instead, they rely on the statistical or structural properties of the image itself to verify its authenticity (Mushtaq and Mir, 2014; Liu et al., 2020; Kaur and Gupta, 2019). Passive techniques are often used when it is not feasible to embed the watermark data into the image or when real-time authentication is required (Wei et al., 2019; Armas Vega et al., 2020; Aldahdooh et al., 2018). However, the passive technique cannot precisely locate the tampering area of the image and does not provide self-recovery.

Active image authentication may provide the self-recovery capability. The technique divides the cover images into non-overlapping blocks. The technique generates the watermark data of each image block comprising the authentication and recovery data. The authentication data is embedded in the original block, while the recovery data is embedded into a different block within the image determined by the block mapping (Dadkhah et al., 2014; Ernawan et al., 2022; Sinhal and Ansari, 2022). The watermarked

* Corresponding author.
E-mail address: afrig@amikom.ac.id (A. Aminuddin).

https://doi.org/10.1016/j.jksuci.2023.101755
1319-1578/

image is then distributed or published into communication channels with possible attacks. The receiver can now authenticate the tampered region of the image using the authentication data and recover it using the recovery data. This recovery capability proves advantageous when the original image is either unavailable or has been extensively disseminated, making requesting the original image from all users impractical (Raj and Shreelekshmi, 2021; Anbu, 2020; Anand and Singh, 2020).

The tamper coincidence problem can arise when the tampered region may coincidentally match both the original blocks and the recovery data, resulting in the inability to recover the tampered area of the image (Tohidi et al., 2021; Huang et al., 2022; Ernawan, et al., 2022). It is essential to carefully design the block mapping technique to mitigate the tamper coincidence problem. The technique may involve a more sophisticated block mapping technique that is less prone to the tamper coincidence problem. Additionally, embedding multiple instances of the recovery data implementing multiple block mapping may be necessary to improve image authentication and self-recovery reliability.

Tai and Liao (Tai and Liao, 2018) presented an image authentication method that detected image tampering and facilitated self-recovery. The technique utilized a chaotic map embedding sequence to insert a delicate watermark comprising authentication and recovery data into another block. The wavelet transform was utilized instead of the average to mitigate the block artifacts in the recovered images, enhancing the image's contrast. A hierarchical tamper detection strategy was implemented to achieve accurate detection of tampering. Simulation results demonstrated that the method could withstand collage and constant-average attacks while maintaining high accuracy in tamper localization. However, using a chaotic map to generate the block map introduced the issue of tamper coincidence despite a low tampering rate, adversely affecting the quality of the recovered image.

Fan and Wang (Fan and Wang, 2018) presented an enhanced fragile watermarking scheme for digital image protection and self-recovery. The scheme utilized the Set Partitioning in Hierarchical Trees (SPIHT) algorithm at a block level to ensure that modifying the output bits of the source encoder would only affect the corresponding image blocks rather than compromising the reconstruction of the entire image. The embedding location of the recovery data was determined using a chaotic map based on the logistic map. The check bits were scrambled using a chaotic sequence, which improved the tampering discrimination capability of the scheme. However, using the logistic map in the block mapping technique may lead to tamper coincidence problems, resulting in a lower-quality recovered image.

Molina-Garcia et al. (Molina-Garcia et al., 2020) presented a fragile watermarking scheme to authenticate and self-recover color images. The original image is divided into non-overlapping blocks, with two watermarks generated for recovery and authentication purposes in each block. Using a permutation process, these watermarks were then embedded into different blocks, occupying the two least significant bits. A bit-adjustment phase was subsequently applied to enhance the quality of the watermarked image. A hierarchical tamper detection algorithm was employed to achieve accurate detection of tampering. To address the issue of tamper coincidence, three recovery watermarks were embedded in different positions to reconstruct a specific block. However, the random block mapping technique based on the permutation process may still lead to instances of tamper coincidence despite a small tampered area, as shown in the experimental result.

Sinhal et al. (Sinhal et al., 2020) presented a blind fragile watermarking technique designed for color images to provide tamper detection and self-recovery capabilities. The scheme employed a secret key-based pseudo-random binary sequence as a fragile watermark for tamper detection, with the recovery information randomly preserved using the same secret key. In the embedding process, each channel of the RGB image was divided into non-overlapping blocks of size 2×4 pixels. Experimental results demonstrated that the scheme successfully identified tampered regions. However, the random distribution of the recovery data introduced the possibility of tamper coincidence occurring despite a low tampering rate, which will reduce the recovered image quality.

Reyes-Reyes et al. (Reyes-Reyes et al., 2021) presented a fragile watermarking scheme to handle high tampering rates and provide color image authentication and self-recovery capabilities. The original image is divided into non-overlapping blocks, generating a recovery watermark for each block. A single bit was derived for the block authentication by applying the bitwise exclusive OR (XOR) operation to the recovery watermarks. To address the issue of tamper coincidence, the embedding and extraction process could be implemented in three variants. Three, six, or nine copies of the generated watermarks could be embedded depending on the chosen variant. In the post-processing stage, a specialized procedure was applied to identify regions affected by the tamper coincidence problem within each recovery watermark. However, the experimental results show that the scheme does not completely eliminate the tamper coincidence problem occurring at a low tampering rate.

Hussan et al. (Hussan et al., 2022) presented an image watermarking technique for the detection of tampering and recovery of color images. The method initially separated the color image into three planes, each further divided into four equal halves. These halves were then subdivided into non-overlapping blocks of size 4×4. From a group of four corresponding sub-blocks, a 32-bit watermark was generated, consisting of the average value and an 8-bit data segment indicating the location of the mapped block. This 32-bit watermark was encrypted using gray code and subsequently embedded into 2LSB of the mapped block. The embedding process utilized a chaotic sequence to ensure a recovery process could be performed even if all three planes were tampered with. However, the chaotic sequence employed in this method for block mapping has the potential to introduce tamper coincidence problems.

Sahu et al. (Sahu et al., 2023) presented a dual image-based reversible fragile watermarking scheme. The scheme embeds two secret bits into each pixel using a pixel readjustment strategy. Through maximal modifications of ± 1 to non-boundary pixels based on watermark data, this technique ensures both reversibility and a triple objective of higher capacity, improved perceptual transparency, and robustness. The scheme could identify and localize the tampered regions, maintaining high accuracy and precision across diverse tampering scenarios while demonstrating resilience against intentional and unintentional attacks (Sahu, 2023). In addition, the scheme implemented blind watermark bit generation using a chaotic system based on the logistic map, improving tamper detection and localization efficiency.

It can be summarized that previous research has not considered that the designed block mapping technique will affect the rate of the tamper coincidence problem. A high tamper coincidence problem can lead to a low-quality recovered image caused by losing its recovery information. This paper improves the previous research of the AuSR1 (Aminuddin and Ernawan, 2022) and AuSR2 (Aminuddin and Ernawan, 2022) for image authentication and self-recovery. The improved technique is called the AuSR3, focusing on developing a new block mapping technique to avoid the tamper coincidence problem. In addition, the AuSR3 also improves the LSB shifting technique to produce a high-quality watermarked image. In summary, the contribution of this paper is presented as follows:

1) Tamper coincidence problem: The new block mapping technique of the AuSR3 is expected to eliminate the tamper coincidence problem by up to 25% tampering rate. In addition, when it is higher than 25%, the AuSR3 is expected to decrease the tamper coincidence problem by up to 10%.
2) Watermarked image quality: The improved LSB shifting algorithm is expected to improve the watermarked image quality by up to 2 dB of PSNR value compared to the LSB replacement technique.
3) Recovered image quality: The elimination of the tamper coincidence problem and the improved LSB shifting algorithm is expected to increase the recovered image quality by up to 2 dB of PSNR value compared to the existing techniques.

In addition, the practical advantages of tamper localization in the AuSR3 are diverse and span various fields, such as digital forensics, criminal investigations, media authentication, biometric security, and medical imaging. Furthermore, self-recovery mechanisms are valuable when tampering attempts might go unnoticed or manual intervention is impractical.

The following sections of the paper are structured as follows: Section 2 reviews related to the existing AuSR framework and the existing block mapping techniques. Section 3 outlines the proposed AuSR3 method, including the AuSR3 block mapping, the watermark embedding, the watermark extraction, and its evaluation technique. Section 4 presents experimental results for the proposed AuSR3 method and compares its performance to the existing methods. Finally, Section 5 offers conclusions on the AuSR3.

2. Related works

This section discusses the existing image authentication and self-recovery framework (AuSR) from the previous research. In addition, this section also presents two block mapping techniques to decide the embedding location of the recovery data: random block mapping and uniform block mapping.

2.1. AuSR framework

The authentication and self-recovery (AuSR) framework was previously introduced in the AuSR1. It covers various basic authentication and self-recovery stages. The framework includes watermark embedding, tamper localization, and self-recovery techniques. The AuSR1 was superior to the existing techniques regarding watermarked image quality, tamper localization accuracy, and recovered image quality. Further improvements were made on the AuSR2, emphasizing the texture preservation technique for recovery. The contribution of the AuSR1 and the AuSR2 is shown in Fig. 1. It also includes the improvement of the AuSR3 in the block map generation, watermark embedding, and block map reconstruction, highlighted in blue.

The main concern of proposing the AuSR3 is that the AuSR1 and AuSR2 still suffer the tamper coincidence problem despite a small tampering area. Further investigation revealed that the random block mapping technique is the cause of this problem. A block map decides the embedding location of the recovery data. If the recovery data is embedded in the original block location, then the recovery data will not be available when the block has been tampered with. In the AuSR1 and AuSR2, the recovery data is embedded into another block location based on the random block mapping technique. In a random block map, the recovery data may be embedded near the original block location, leading to the tamper coincidence problem when both blocks are tampered with. Therefore, the AuSR3 is proposed to avoid the tamper coincidence problem, eventually increasing the recovered image quality.

2.2. Random block mapping

A random block map can be produced using the Pseudo-Random Number Generator (PRNG). PRNG is a widely used computational tool that produces a sequence of numbers that appears to be random but is generated by a deterministic algorithm. The sequence of numbers produced by a PRNG is not truly random because it is based on a mathematical formula or algorithm. It means that the output is predictable and can be replicated if the seed value and algorithm used to generate it are known. This deterministic behavior is an advantage in block map generation as it generates identical block maps in the watermark embedding and extraction process. The AuSR1 and AuSR2 defined the seed value as the key for authentication and self-recovery. Even though PRNG has a deterministic behavior, the output of a good PRNG will be statistically random, meaning that it will have properties similar to a truly random sequence of numbers, such as being uniformly distributed and having no discernible patterns. In a random block mapping technique, the original block location is randomized using PRNG and a secret key to produce a random block map. The random block map can be illustrated in Fig. 2.

Fig. 2(a) shows that each image block is numbered from 1 to n, where n is the maximum number of image blocks. In the example, the 1D image block has eight blocks, while the 2D image block has 64 image blocks. The numbered block is then scrambled using PRNG to produce a random block map, shown in Fig. 2(b). This random block map determines the embedding location of the recovery data in AuSR1 and AuSR2. This random block map is the basic technique to prevent the tamper coincidence problem. Furthermore, Fig. 2(c) shows the Euclidean distances between the original block location and the random block map. The Euclidean distance can be calculated as follows:

$$ed_{x,y} = \sqrt{(ob_x - rb_x)^2 + (ob_y - rb_y)^2} \quad (1)$$

where ob_x and ob_y represent the original block location, rb_x and rb_y denote the recovery block location, and $ed_{x,y}$ indicates the Euclidean distance of a block with $\times$ and y coordinates. In Fig. 2, the Euclidean distance is color-coded between red and yellow, representing its probability of tamper coincidence problem. Red means a high risk of tamper coincidence problem, while yellow indicates a low risk of tamper coincidence problem. Furthermore, the distribution of the Euclidean distance in a random block map is shown in Fig. 3.

Fig. 3 shows the random block map distribution based on 256 $\times$ 256 blocks. It is based on an image with a size of 512 $\times$ 512 pixels divided into blocks of 2 $\times$ 2 pixels. The bell-shaped curve of the normal distribution can be seen up to the Euclidean distance of 256, which is the maximum width and height of the image block. When the Euclidean distance passes this value, the distribution slopes down to $\sqrt{2}$ of 256, representing the maximum Euclidean distance of the image diagonal. Based on Fig. 3(b), it can be seen that almost 30% of the blocks lie at a high risk of tamper coincidence problems since the recovery block is embedded near the original block location. In addition, 75% of recovery data is embedded in the medium to high risk of tamper coincidence problem.

2.3. Uniform block mapping

As previously mentioned, the random block mapping technique has a limitation in which many recovery data are embedded near the original block locations. It raises the question of the furthest distance possible to embed the recovery data. In a 1D block map,

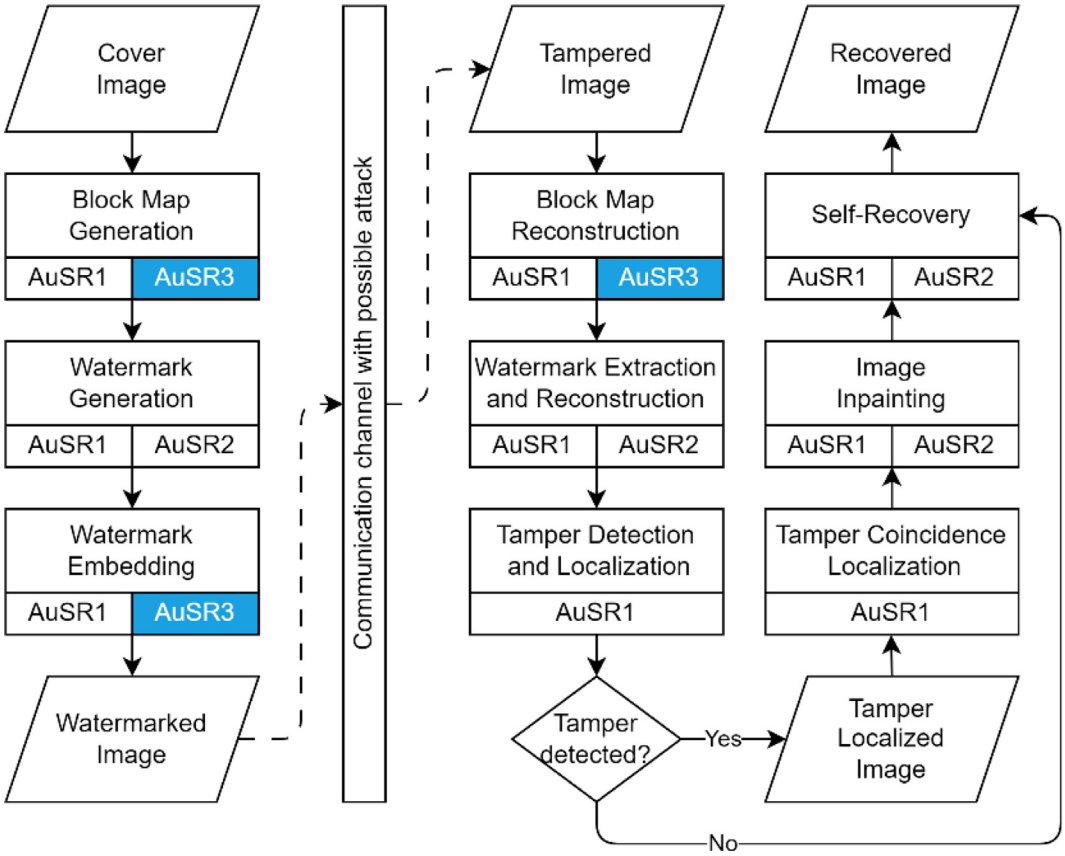

Fig. 1. Image authentication and self-recovery framework.

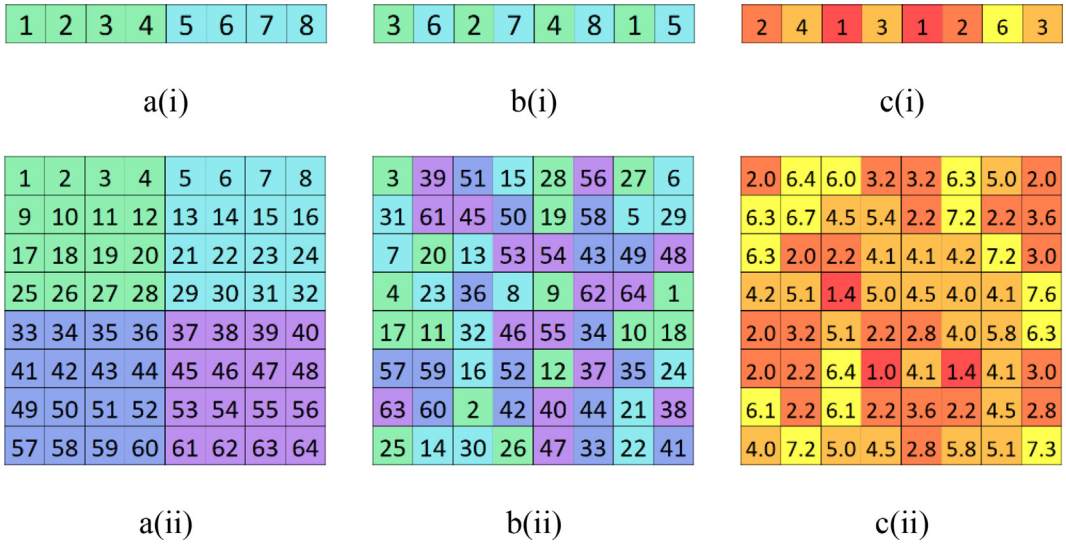

Fig. 2. Random block map and its Euclidean distance (a) Original block location (b) Random block map(c) Euclidean distance (i) 1D block map (ii) 2D block map.

the recovery data of the first block can be embedded into the last block. Next, the recovery data of the second block can be embedded into the second last block. It ensures that the first block has the most distance recovery data location. However, the recovery data of the center block will be embedded into the next adjacent block, which will cause the tamper coincidence problem. A better way to embed the recovery data is based on uniform block mapping. It embeds the first half of the block map into the second half sequentially. Thus, each recovery data is embedded in an equal distance, the half-width of the 1D block map. The uniform block map can be illustrated in Fig. 4.

Based on Fig. 4(c), the Euclidean distance between the original block location and the recovery data in a 1D block map is exactly four blocks, half the image width. In comparison, a 2D uniform block map embeds the recovery data based on the following equation:

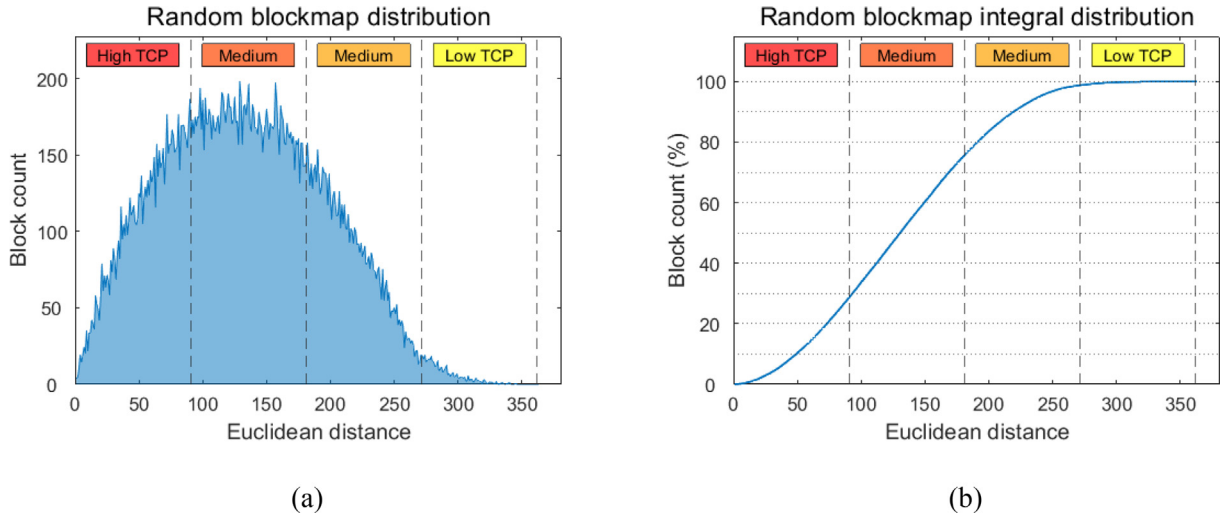

Fig. 3. Random block map distribution (a) Distribution (b) Integral distribution.

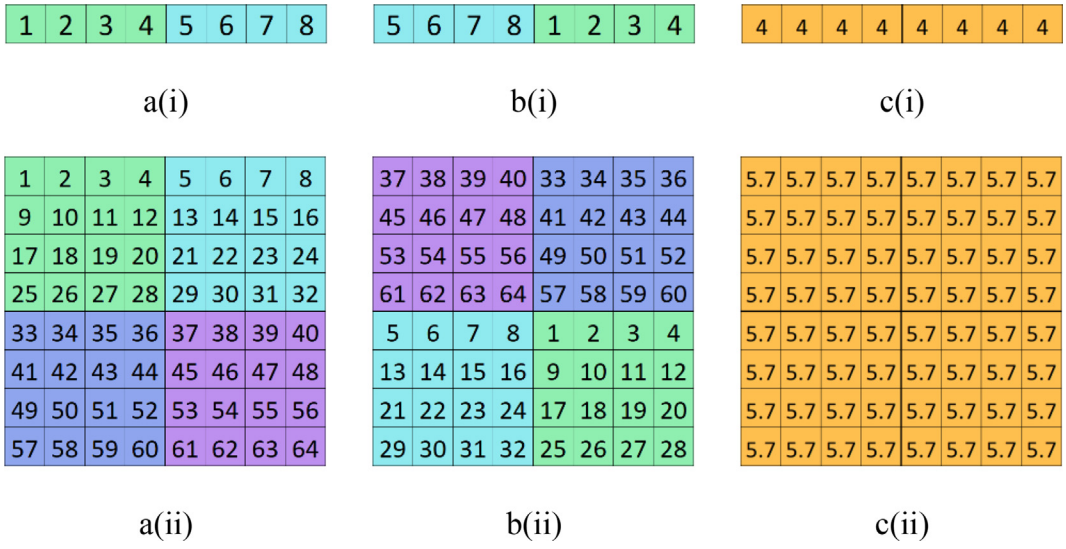

Fig. 4. Uniform block map and its Euclidean distance (a) Original block location (b) Uniform block map(c) Euclidean distance (i) 1D block map (ii) 2D block map.

$$rb_x = \lfloor ob_x + \left(\frac{wb}{2}\right) \rfloor \bmod wb \tag{2}$$

$$rb_y = \lfloor ob_y + \left(\frac{hb}{2}\right) \rfloor \bmod hb \tag{3}$$

where wb and hb represent the width and the height of the block map, $ob_{x,y}$ denotes the original block location, $rb_{x,y}$ indicates the recovery block location, and $\lfloor x \rfloor$ represents the floor function of x. The uniform block map provides a medium risk of tamper coincidence problems. However, from a security perspective, it is easy for the attacker to locate the recovery data since the distribution is predictable. In addition, the uniformity of the block map may produce a large chunk of tamper coincidence problems when the tampering rate is larger than 25%. Therefore, it will reduce the quality of the recovered image.

3. Proposed method

This section explains the proposed method of the AuSR3. At first, the new block mapping technique of the AuSR3 is presented. Next, the watermark embedding and extraction are discussed. Finally, the evaluation of the AuSR3, including the TCP rate, precision, PSNR, and SSIM, is explained.

3.1. AuSR3 block mapping

The proposed AuSR3 block mapping technique aims to overcome the limitation of random and uniform block mapping while maintaining the advantages. The random block mapping technique has the advantage of the security key involved in PRNG, but it may produce a high risk of tamper coincidence problems within a small Euclidean distance. In comparison, the uniform block mapping technique has the advantage of a safe Euclidean distance of recovery data, while it does not provide any security. The proposed AuSR3 block map is generated as follows:

1) Compute the required size of the block map. The block map size is $M/2 \times N/2$, where M represents the image's height, and N indicates the image's width.
2) Generate a vector rn to store random numbers from 1 to n, where n represents the number of blocks in Step 1. The random number is generated using the PRNG with an integer security key, as shown in (4).

3) Generate a matrix for the AuSR3 block map with the required size as in Step 1. Populate the block map using (5) and (6) starting from the central location of the block map. It moves in an outward spiral direction until it reaches the edge of the block map.

$$rn_{1\cdots n} = PRNG(n, key) \tag{4}$$

$$rb_x = \begin{cases} \lfloor ob_x + (\frac{wb}{2}) \rfloor \bmod wb, & \text{if}, rn_i \bmod 2 = 0 \\ rn_i \bmod wb, & \text{if}, rn_i \bmod 2 = 1 \end{cases} \tag{5}$$

$$rb_y = \begin{cases} \lfloor ob_y + (\frac{hb}{2}) \rfloor \bmod wb, & \text{if}, rn_i \bmod 2 = 1 \\ rn_i \bmod wb, & \text{if}, rn_i \bmod 2 = 0 \end{cases} \tag{6}$$

where wb represents the width of the block map, hb denotes the height of the block map, $ob_{x,y}$ denotes the original block location, and $rb_{x,y}$ indicates the recovery block location. The equation shows that the recovery data is embedded into a random x-axis with a fixed y-axis or a fixed x-axis with a random y-axis. The fixed axis is derived from the uniform block mapping technique, while the random axis is derived from the random block mapping technique.

4) Resolve the conflict that may occur when the $rb_{x,y}$ is already populated in the previous iteration. Increment the value of rn_i until it reaches the increment threshold. This threshold is defined to prevent an infinite loop.
5) List all the residual conflicts in a vector. The 1D uniform block mapping technique resolves the residual conflicts until all blocks are mapped accordingly.

Once the block map is completed, it can be used for the watermark embedding and extraction process. The block map reconstruction process must use the identical security key to the block map generation process. Different security keys will produce different block maps, making it unusable for self-recovery. The proposed AuSR3 block mapping technique can be illustrated in Fig. 5.

The original block location is color-coded to show each region of the block map, as shown in Fig. 5(a). In the 1D block map, the block map is divided into two regions, while in the 2D block map, the block map is divided into four regions. Fig. 5(b) shows that the recovery data is mapped in a different region of the block map. For example, the purple region of the block map is distributed randomly to the other three regions. Therefore, the minimum Euclidean distance of the 2D block map is 4, representing half of the image's width. Furthermore, the comparison of block map distribution is shown in Fig. 6.

In Fig. 6(a), the block map size is 256 × 256 blocks, representing the number of image blocks tested in the experiment. It shows that the block map distribution of the proposed AuSR3 is shifted to the right compared to the random block map distribution. In comparison, the uniform block map has 65.536 blocks, with a single Euclidean distance value being half the block map diagonal distance. In Fig. 6(b), 47% of the random block map lies within the Euclidean distance of less than 128, representing half of the image width or height. Therefore, these blocks may suffer the tamper coincidence problem despite a low tampering rate of less than 25%. In comparison, the minimum Euclidean distance of the AuSR3 block map is 128, which prevents the tamper coincidence problem on the image when the tampering rate is less than 25%. Furthermore, each block mapping technique will be evaluated to find the tamper coincidence problem rate under various tampering scenarios.

3.2. Watermark embedding

The proposed method embeds the watermark data into two LSB of 2 × 2 pixels image block. The watermark data consists of authentication data and recovery data. The authentication data is designed to be sensitive to changes in the image. Thus, when embedded, any tampering or modification of the image would cause the authentication data to become distorted or destroyed, which makes it possible to detect any unauthorized changes made to the image. On the other hand, the recovery data is embedded into another block location based on the block mapping to prevent the recovery data from being destroyed when tampering or modification occurs. The embedding process is shown as follows:

1) Divide the cover image into RGB channels and divide each channel into image blocks of 2 × 2 pixels.
2) Generate the proposed AuSR3 block map with a secret integer as a key for the selected channel. The block map size and the channel type (red = 1, green = 2, or blue = 3) are incorporated into the secret key to ensure each channel has a different AuSR3 block map.

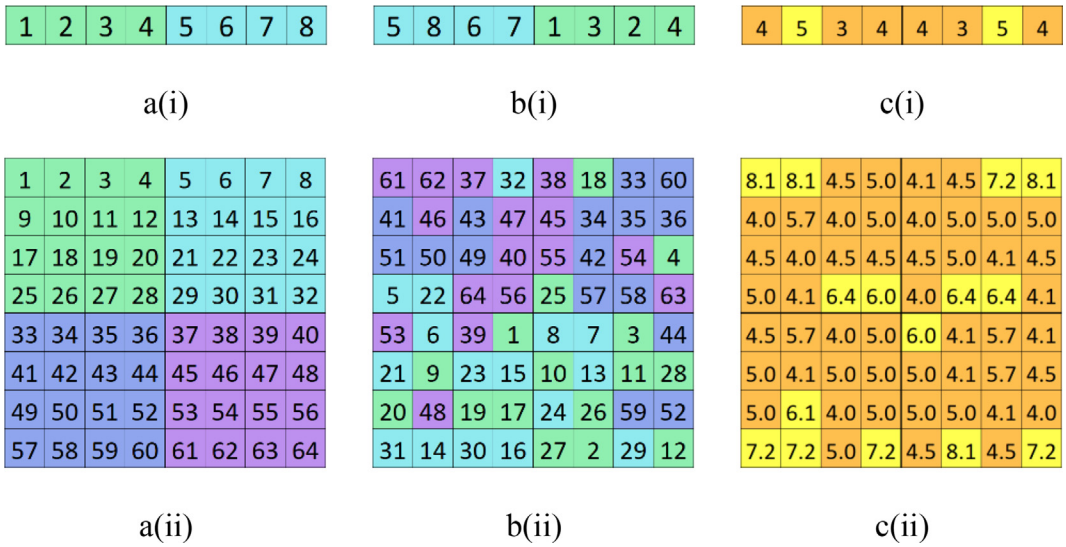

Fig. 5. AuSR3 block map and its Euclidean distance (a) Original block location (b) AuSR3 block map(c) Euclidean distance (i) 1D block map (ii) 2D block map.

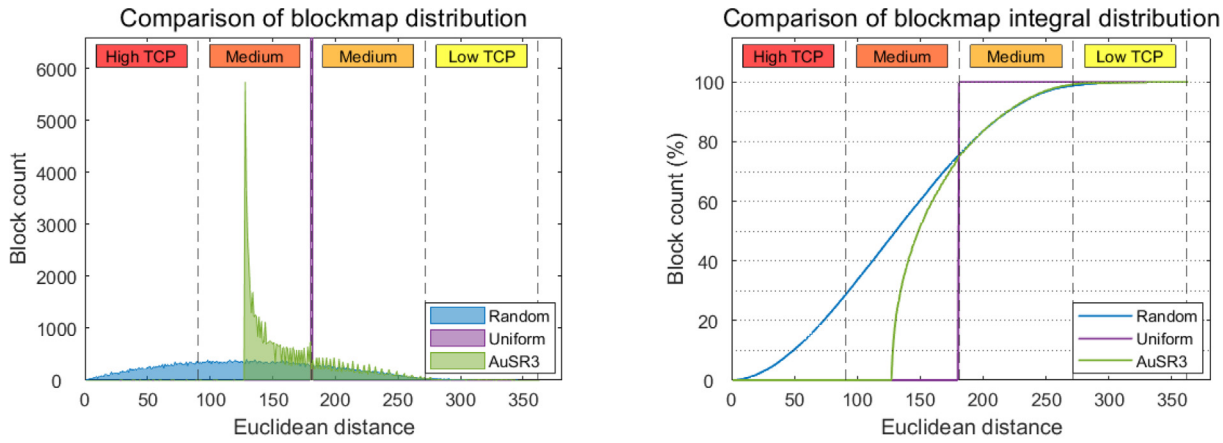

Fig. 6. Comparison of block map distribution (a) Distribution (b) Integral distribution.

3) Calculate the average value of the four pixels of each image block and take 6 MSB of the average value as the recovery data.
4) Embed the recovery data into a distant block location based on the AuSR3 block map. Select three pixels of the target block location to embed the 6 bits of recovery data using the LSB shifting algorithm (Aminuddin and Ernawan, 2022). Those three pixels are randomly selected using the target block map location as the secret key.
5) Generate two bits of authentication data from the first 6 MSB of the selected three pixels in the original block location. The first bit is taken from the parity bit of those 18 bits. At the same time, the second bit is taken from the parity bit of the target block map location. This step ensures that two similar blocks would have different authentication data.
6) Embed the authentication data into the last pixel of its original block location using the LSB shifting algorithm (Aminuddin and Ernawan, 2022). It is safe to embed the authentication data using the LSB shifting algorithm since the selected pixel is not considered in the authentication data generation.
7) Repeat steps 2 to 6 until each image channel and block are embedded with the respective watermark data.

Once the authentication and recovery data is successfully embedded, the watermarked image can be published or sent to the internet. If the watermarked image undergoes any modification, the proposed method should be able to detect and localize the tampered region of the image. In addition, the tampered region could be recovered using the recovery data.

3.3. Watermark extraction

In the communication channel, the watermarked image may be tampered with or modified by an unauthorized party. The proposed method can extract the watermark data to localize and recover the tampered region of the image. The extraction process consists of block map reconstruction, watermark extraction and reconstruction, tamper detection and localization, tamper coincidence localization, image inpainting, and self-recovery. The extraction process is described as follows:

1) Divide the tampered image into RGB channels and divide each channel into image blocks of 2×2 pixels.
2) Reconstruct the AuSR3 block map of each channel using the secret key. The secret key is used as the seed number of the PRNG in the proposed AuSR3 block mapping technique. Thus, the reconstruction process can only be done using the identical key as the block map generation process.
3) Extract the watermark data from the tampered image. The process extracts two LSB of the selected image block. The first two bits are the authentication data of the selected image block, while the last six bits are the recovery data of another block based on the inversed AuSR3 block map.
4) Reconstruct the authentication and recovery data from the tampered image. The reconstruction process is identical to the watermark generation process. The only difference is that the watermark generation process takes the cover image as the input, while the watermark reconstruction process takes the tampered image as the input.
5) Localize the tampered region of the image using three-layer authentication, previously defined in (Aminuddin and Ernawan, 2022). The three-layer authentication requires two inputs: the extracted and reconstructed authentication data.
6) Localize the tamper coincidence problem inside the tampered region of the image. A block is considered a tamper coincidence problem when the original and recovery blocks are tampered with at the same time.
7) Solve the tamper coincidence problem using the image inpainting as follows:

$$tc = \frac{\sum_{i=1}^{8} ntc_i \cdot w_i}{\sum_{i=1}^{8} w_i} \tag{7}$$

$$d_i = \sqrt{(ntc_x - tc_x)^2 + (tc_y - tc_y)^2}, i = 1 \cdots 8 \tag{8}$$

$$w_i = \left(1 - \frac{d_i}{d_{max}}\right) \cdot d_i^{-2} \tag{9}$$

where tc and $tc_{x,y}$ represents the new pixel value and location of the tamper coincidence problem, ntc_i and $ntc_{x,y}$ describes the current pixel value and location of the selected neighboring non-tamper coincidence problem, d_i indicates the Euclidean distance between $tc_{x,y}$ and $ntc_{x,y}$, and w_i is the weight of the selected ntc_i. Each $ntc_{x,y}$ is the closest neighbor of $tc_{x,y}$, representing the cardinal direction within a 45-degree angle.

8) Recover the tampered blocks with their respective recovery data. If the recovery data undergoes the tamper coincidence problem, the output of the image inpainting is used instead.
9) Repeat steps 2 to 8 until the tampered region of each image channel and block are recovered.

Once the extraction process is completed, the proposed method produces three images. The first image shows the tampered region of the image in white, while the untampered region is shown in black. The second image is the recovered image. The third image shows the tamper coincidence problem within the recovered image.

3.4. Evaluation

The proposed method can be investigated through three evaluation techniques. The performance of the block mapping technique can be evaluated based on the rate of the tamper coincidence problem. The precision of the tamper localization technique can be evaluated using the confusion matrix. The watermarked and recovered image quality can be evaluated using the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM). The rate of the tamper coincidence problem is defined as follows:

$$tcp_{rate} = \frac{tcp}{m \cdot n} \tag{10}$$

where tcp represents the number of pixels with the tamper coincidence problem, m represents the image's height, and n indicates the image's width. A low tcp_{rate} value represents a robust block mapping technique, while a high tcp_{rate} value represents severe tamper coincidence problems. The precision of the tamper localization is calculated as follows:

$$precision = \frac{TP}{TP + FP} \tag{11}$$

where TP represents the number of true positive detections, and FP represents the number of false positive detections. Finally, the watermarked and recovered image quality is measured using the PSNR (Pourasad et al., 2021). The PSNR is defined by:

$$PSNR(p,q) = 10 \log_{10} \frac{S^2}{\frac{1}{m \cdot n} \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} (p(x,y) - q(x,y))} \tag{12}$$

where S represents the maximum intensity value of 255 on an 8-bit image, p is the cover image, and q is the watermarked or recovered image. PSNR is calculated by comparing the mean squared error (MSE) between the two images. The MSE represents the average squared difference between both images in the corresponding pixel locations. PSNR is derived from MSE and expressed in decibels (dB), which provides a logarithmic scale to represent the difference in quality. Higher PSNR values indicate lower distortion levels and better quality, while lower values indicate more pronounced distortion and poorer quality. Furthermore, SSIM is also used to measure the similarity between two images based on the quality perception of the Human Visual System (HVS) (Ranjbarzadeh et al., 2020). The SSIM is defined by:

$$SSIM(p,q) = \frac{2\mu_p\mu_q + C_1}{\mu_p^2 + \mu_q^2 + C_1} \cdot \frac{2\sigma_p\sigma_q + C_2}{\sigma_p^2 + \sigma_q^2 + C_2} \cdot \frac{\sigma_{pq} + C_3}{\sigma_p\sigma_q + C_3} \tag{13}$$

where $C1$, $C2$, and $C3$ are numerical constants that stabilize the division of a weak denominator. SSIM compares the local image structure in small windows across two images. It measures the similarity of the luminance, contrast, and structure of each window between the two images and combines these measures to obtain an overall SSIM score. SSIM is a perceptual metric used to measure the similarity between two images. Unlike PSNR, which primarily focuses on pixel-level differences, SSIM considers structural information and human visual perception systems. The SSIM index ranges from −1 to 1, where 1 indicates perfect similarity between the images, and −1 indicates complete dissimilarity. The closer the SSIM value is to 1, the more similar the images are perceived.

4. Experimental results

In this section, multiple experiments are performed to evaluate the superiority of the proposed method compared to the existing methods. The experiments were carried out on a computer with an 8 × 2 cores of 1.8 GHz base clock AMD Ryzen 7 5700U and 32 GB memories which runs Matlab R2021a on the Windows 11 operating system. The test images are taken from the USC-SIPI database ("SIPI Image Database, 2023) comprising eight color images: Airplane, Baboon, House, Lena, Peppers, Sailboat, Splash, and Tiffany. Each image has a size of 512 × 512 pixels. In addition, two more datasets are included in the experiment: Kodak-PCD0992 and UCID-1338. The Kodak-PCD0992 dataset ("True Color Kodak Images, 2023) comprises 24 images (512 × 768 pixels), while the UCID-1338 dataset (Schaefer and Stich, 2004) comprises 1.338 images (512 × 384 pixels). The performance of AuSR3 is assessed in four sets of experiments. The first set compares the watermarked image quality using PSNR and SSIM between the cover and watermarked images. The second set compares the precision of the tamper detection and localization. The third set compares the number of the tamper coincidence problem. The fourth set compares the recovered image quality in terms of PSNR and SSIM between the cover and the recovered image.

4.1. Watermarked image quality

In the first set of experiments, the watermark data consisting of authentication and recovery data are embedded into the cover image. The differences between the cover and watermarked images can be measured using PSNR and SSIM. As previously mentioned, the AuSR3 embeds the cover image with the watermark data using the LSB shifting algorithm improved from AuSR1 and AuSR2. Previously, the LSB shifting algorithm could only be used for embedding the recovery data, while the authentication data was embedded by replacing 2 LSB of the selected pixel. Implementing the LSB shifting algorithm on authentication data may render the authentication data useless since the authentication data generation was computed based on 6 MSB of all pixels in a block. This is because the LSB shifting algorithm may modify up to 6 MSB of the selected pixel for embedding the authentication data. Therefore, the authentication data of that block must be recomputed to produce new authentication data. However, embedding the new authentication data using the LSB shifting algorithm may further destroy the new authentication data. It will lead to an infinite loop or race condition unless the authentication data is embedded using 2 LSB replacements.

The improvement of AuSR3 is that all the watermark data can be embedded using the LSB shifting algorithm, including the recovery data and authentication data. The AuSR3 only considers the n-1 pixels in each block for authentication data generation. The authentication data generation does not consider the selected pixel for authentication data embedding. Therefore, the LSB shifting algorithm can embed all the watermark data without destroying the authentication data. As a result, the quality of the watermarked image is significantly improved, as shown in Tables 1–3.

The AuSR3 achieves an average PSNR value of 46.20 dB and SSIM value of 0.9978 in the USC-SIPI dataset, higher than the existing schemes. In comparison, the most common technique for LSB embedding is using the LSB replacement technique, which has

been implemented by Tai (Tai and Liao, 2018) and Fan (Fan and Wang, 2018). In this technique, the difference between the original and watermarked pixels ranges from 0 to 3 intensity levels for 2 LSB embedding. When these differences are computed based on PSNR, the value will be averaged to 44.08 dB. In comparison, Molina-Garcia (Molina-Garcia et al., 2020) and Hussan (Hussan et al., 2022) have implemented the bit adjustment technique, which produces the PSNR value of 44.64 and 44.83 dB, respectively. Their technique has slightly improved the common LSB replacement technique. However, the AuSR3 significantly improves the watermark embedding technique using the LSB shifting algorithm, producing a high-quality watermarked image compared to the existing techniques.

4.2. Tamper detection and localization

In the second set of experiments, the watermarked images are tampered with using regular and irregular attacks. Regular attacks refer to adding a square noise in the central region of the watermarked images, ranging from 10% to 80% tampering rates. Regular attacks provide precise tampering rates, while irregular attacks provide a real-life example of image tampering attacks. Once the image is tampered with, the AuSR3 can be used to detect and localize the tampered region of the image. The tamper localization technique can be evaluated based on the confusion matrix comprising true positive rate (TPR), false positive rate (FPR), false negative rate (FNR), and true negative rate (TNR). Furthermore, the precision, F1 score, and accuracy can be derived from the confusion matrix. These values are then compared to the existing schemes to show the performance of tamper localization, as shown in Table 4 and Table 5.

Table 4 shows that the AuSR3 can produce high precision, F1 score, and accuracy. Furthermore, Table 5 shows that the AuSR3 is comparable to the AuSR1 and produces a high precision compared to other techniques. The result of AuSR1 was quite similar to AuSR3 because both use the same technique with block sizes of 2×2 pixels, while AuSR2 employed a larger block size of 3×3 pixels. Smaller block sizes increase the precision of tamper localization as it can reduce the FPR. However, the FPR can be tolerated in image authentication and self-recovery as the false positive detection should be recovered in the recovery process. At the same time, FNR will render the tampered region of the image unrecoverable as the recovery process fails to detect the block as a tampered block. Therefore, researchers are competing to maximize the TPR while minimizing the FNR. Furthermore, the authentication data for each block is limited to 2 bits. If computed based on the probability, 1-bit authentication data may produce 50% TPR, while 2-bit authentication data may produce 75% TPR. The AuSR3 implemented the three-layer authentication algorithm, increasing the TPR to 100% on regular attacks and 99% on irregular attacks. Therefore, it increases the chance of recovering the tampered region.

The AuSR3 embeds the authentication data on the randomly selected pixel of the image block. One of the four pixels in the image block stores the authentication data, while the other three are used to store the recovery data of another block. The image tampering attack on an image block can be categorized into three possible scenarios. The first scenario is that the tampering attack only applied to the pixel with the authentication data. The second scenario is that the tampering attack only occurred on the pixels without the authentication data. In both cases, the difference between the extracted and reconstructed authentication data will firmly localize the block as the tampered block. The third scenario is that all the pixels in the image block are tampered with. In such a case, the extracted and reconstructed authentication data could be precisely identical, which leads to false negative detection. Therefore, the three-layer authentication is implemented to reduce false negative detection.

In Fig. 7, the images from the USC-SIPI dataset undergo various irregular attacks. It includes a Gaussian blur attack applied to the Airplane and Sailboat images. The cropping attacks are applied to the Lena and Tiffany images. Furthermore, various attacks, such as normal, protocol, collage, copy-move forgery, and vector quantization (Bolourian Haghighi et al., 2019), are also applied to the dataset. In this scenario, the AuSR3 can detect and localize the tampered region of the image with high precision and accuracy. In addition, Fig. 7 also shows the tamper coincidence problem under various irregular attacks. The tamper coincidence problem is eliminated when the tamper region is below 25 %, such as on the Splash image.

4.3. Tamper coincidence problem

In the third set of experiments, the performance of the AuSR3 is evaluated based on the rate of the tamper coincidence problem in the tampered region of the image. As mentioned earlier, the AuSR3 implements a new block mapping technique to avoid the tamper coincidence problem. The AuSR3 ensures that the recovery data of a block should be embedded into the most distant location possible from the original block location. Therefore, it lowers the chance of the original and recovery blocks being tampered with at the same time, avoiding the tamper coincidence problem. In this experiment, the watermarked images are tampered with in various locations, ranging from 10% to 80% tampering rates, as shown in Fig. 8.

There are nine regions selected in this experiment: top left, top center, top right, center left, center, center right, bottom left, bottom center, and bottom right. These nine locations are selected to prove that the new block mapping technique is carefully designed to avoid the tamper coincidence problem wherever the tamper is located. Once all of the images are tampered with, the AuSR3 then detects and localizes the tampered region of the images. The tamper localization and the new block mapping are

Table 1
Watermarked images comparison of the PSNR values on the USC-SIPI dataset.

Image	Tai (Tai and Liao, 2018)	Fan (Fan and Wang, 2018)	Molina-Garcia (Molina-Garcia et al., 2020)	Hussan (Hussan et al., 2022)	AuSR1 (Aminuddin and Ernawan, 2022)	AuSR2 (Aminuddin and Ernawan, 2022)	AuSR3
Airplane	44.12	44.11	44.69	44.70	45.68	46.05	**46.37**
Baboon	44.14	44.12	44.64	44.92	45.70	46.06	**46.37**
House	44.18	44.18	44.66	44.70	45.69	46.07	**46.36**
Lena	44.12	44.13	44.60	44.98	45.71	46.06	**46.37**
Peppers	44.06	44.06	44.54	44.76	45.54	45.87	**46.16**
Sailboat	44.11	44.10	44.61	44.87	45.68	46.04	**46.35**
Splash	44.09	44.08	44.47	44.84	45.57	45.93	**46.22**
Tiffany	43.85	43.84	44.87	44.89	44.95	45.20	**45.38**
Average	44.08	44.08	44.64	44.83	45.57	45.91	**46.20**

Table 2
Watermarked images comparison of the SSIM values on the USC-SIPI dataset.

Image	Tai (Tai and Liao, 2018)	Fan (Fan and Wang, 2018)	Molina-Garcia (Molina-Garcia et al., 2020)	Hussan (Hussan et al., 2022)	AuSR1 (Aminuddin and Ernawan, 2022)	AuSR2 (Aminuddin and Ernawan, 2022)	AuSR3
Airplane	0.9781	0.9781	0.9812	0.9830	0.9889	0.9901	**0.9914**
Baboon	0.9941	0.9941	0.9947	0.9984	0.9990	0.9991	**0.9992**
House	0.9815	0.9815	0.9834	0.9950	0.9967	0.9970	**0.9974**
Lena	0.9820	0.9820	0.9840	0.9989	0.9993	0.9994	**0.9995**
Peppers	0.9791	0.9791	0.9816	0.9988	0.9991	0.9992	**0.9992**
Sailboat	0.9868	0.9867	0.9884	0.9968	0.9980	0.9982	**0.9984**
Splash	0.9696	0.9695	0.9737	0.9975	0.9983	0.9985	**0.9987**
Tiffany	0.9805	0.9804	0.9846	0.9978	0.9985	0.9986	**0.9987**
Average	0.9815	0.9814	0.9840	0.9958	0.9972	0.9975	**0.9978**

Table 3
Watermarked images comparison between the AuSR3 and the existing methods across various datasets.

Dataset	AuSR1 (Aminuddin and Ernawan, 2022)		AuSR2 (Aminuddin and Ernawan, 2022)		AuSR3	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
USC-SIPI	45.57	0.9972	45.91	0.9975	**46.20**	**0.9978**
Kodak-PCD0992	45.58	0.9953	45.93	0.9958	**46.22**	**0.9963**
UCID-1338	45.62	0.9949	45.96	0.9954	**46.25**	**0.9958**
Average	45.59	0.9958	45.93	0.9962	**46.22**	**0.9966**

Table 4
The tamper detection and localization of the AuSR3 on the USC-SIPI dataset.

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

Table 5
The precision between the AuSR3 and the existing schemes on the USC-SIPI dataset.

TR	Tai (Tai and Liao, 2018)	Fan (Fan and Wang, 2018)	Molina-Garcia (Molina-Garcia et al., 2020)	Reyes-Reyes (Reyes-Reyes et al., 2021)	AuSR1 (Aminuddin and Ernawan, 2022)	AuSR2 (Aminuddin and Ernawan, 2022)	AuSR3
10	0.9670	0.8007	0.9152	0.9157	1.0000	0.9986	1.0000
20	0.9855	0.9210	0.9580	0.9585	0.9978	0.9934	0.9978
30	0.9903	0.9144	0.9716	0.9718	0.9939	0.9909	0.9939
40	0.9939	0.9483	0.9797	0.9799	0.9918	0.9959	0.9918
50	1.0000	1.0000	0.9884	0.9885	1.0000	0.9890	1.0000
60	0.9943	0.9601	0.9848	0.9849	0.9925	0.9925	0.9925
70	0.9958	0.9748	0.9876	0.9877	0.9787	0.9785	0.9787
80	0.9963	0.9659	0.9891	0.9892	1.0000	0.9500	1.0000
Average	0.9904	0.9357	0.9718	0.9720	0.9943	0.9861	0.9943

then employed to locate the tamper coincidence problem, as visualized in Fig. 9.

In the recovery process, the tamper coincidence problem will be recovered using the image inpainting technique. It solves the tamper coincidence problem using the neighboring pixels to predict the recovery data. The closer the neighboring pixels, the better the recovered image quality, and it will take less time to recover the tamper coincidence problem. In the random block mapping, the tamper coincidence problem is spread throughout the whole tampered region of the image. In comparison, the tamper coincidence problem of the uniform block mapping concentrates on four edges of the tampered region of the image. This high density of tamper coincidence problem makes it difficult for the image inpainting technique to predict the recovery data precisely. In addition, it will take more computation time to solve the tamper coincidence problem. From the security perspective, the recovery data location of the uniform block mapping is highly predictable. Thus, the security of image authentication will be compromised. The comparison of the tamper coincidence problem is shown in Table 6.

Most experiments show that the AuSR3 produces zero tamper coincidence problems on 10% and 20% tampering rates. Theoretically, the new block mapping technique should avoid the tamper coincidence problem by up to 25% tampering rates, as formulated in Eqs. (5) and (6). In comparison, the AuSR2 implemented a random block mapping technique and multiple recovery data, producing 5.27% tamper coincidence problems under a 20% tampering rate. At the same time, the AuSR1 produces a more severe tamper

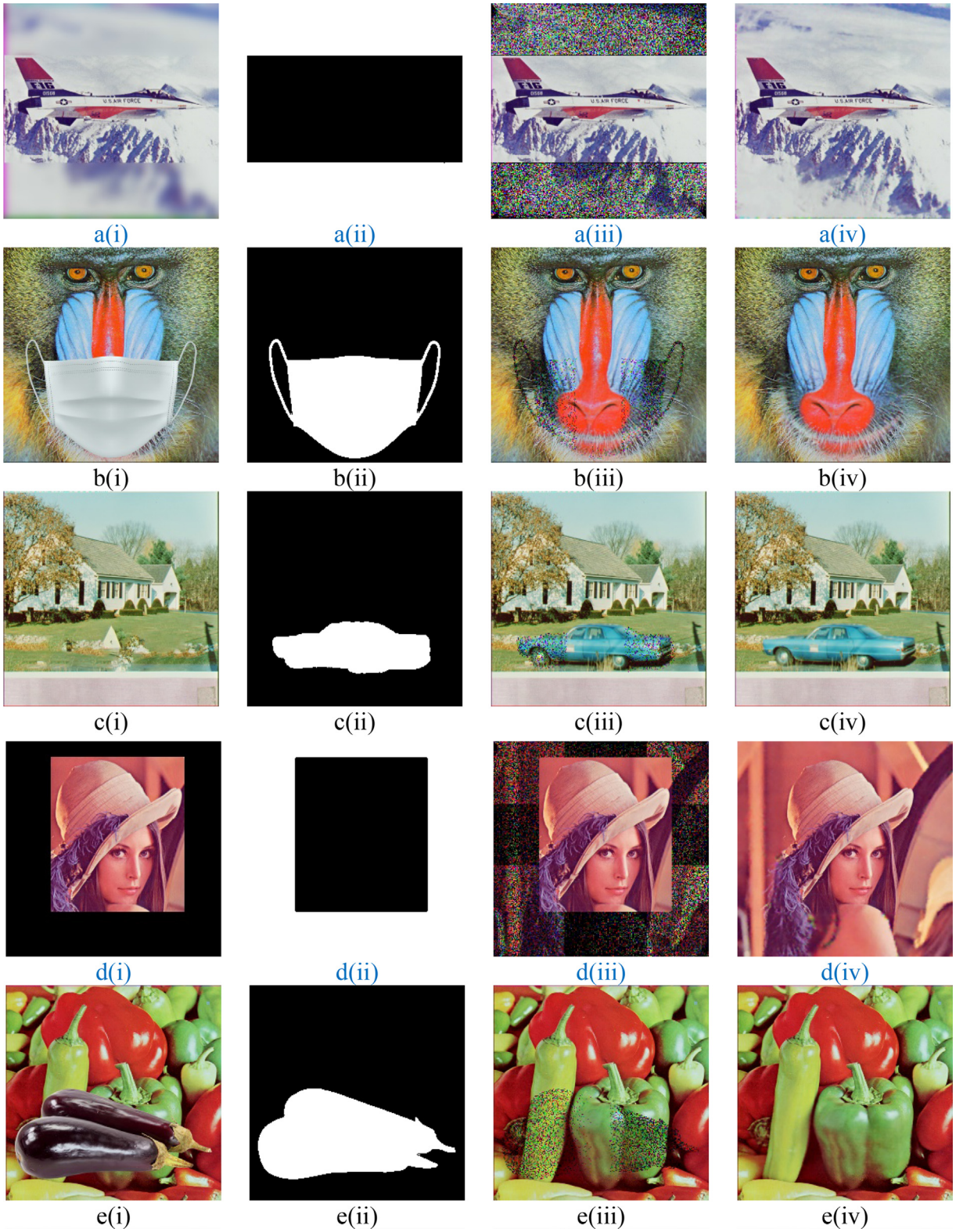

Fig. 7. Irregular tampering attacks (a) Airplane (b) Baboon (c) House (d) Lena (e) Peppers (f) Sailboat (g) Splash (h) Tiffany (i) Tampered image (ii) Tamper detection and localization (iii) Tamper coincidence problem (iv) Recovered image.

coincidence problem of 7.36% on the same tampering rate. It demonstrates that the new block mapping technique outperforms the existing random block mapping techniques to avoid the tamper coincidence problem. Furthermore, the standard deviation of the AuSR3 between multiple tampering locations maximized at 0.55% at a 40% tampering rate. It shows that the tampering locations do not significantly affect the rate of tamper coincidence problem of the AuSR3.

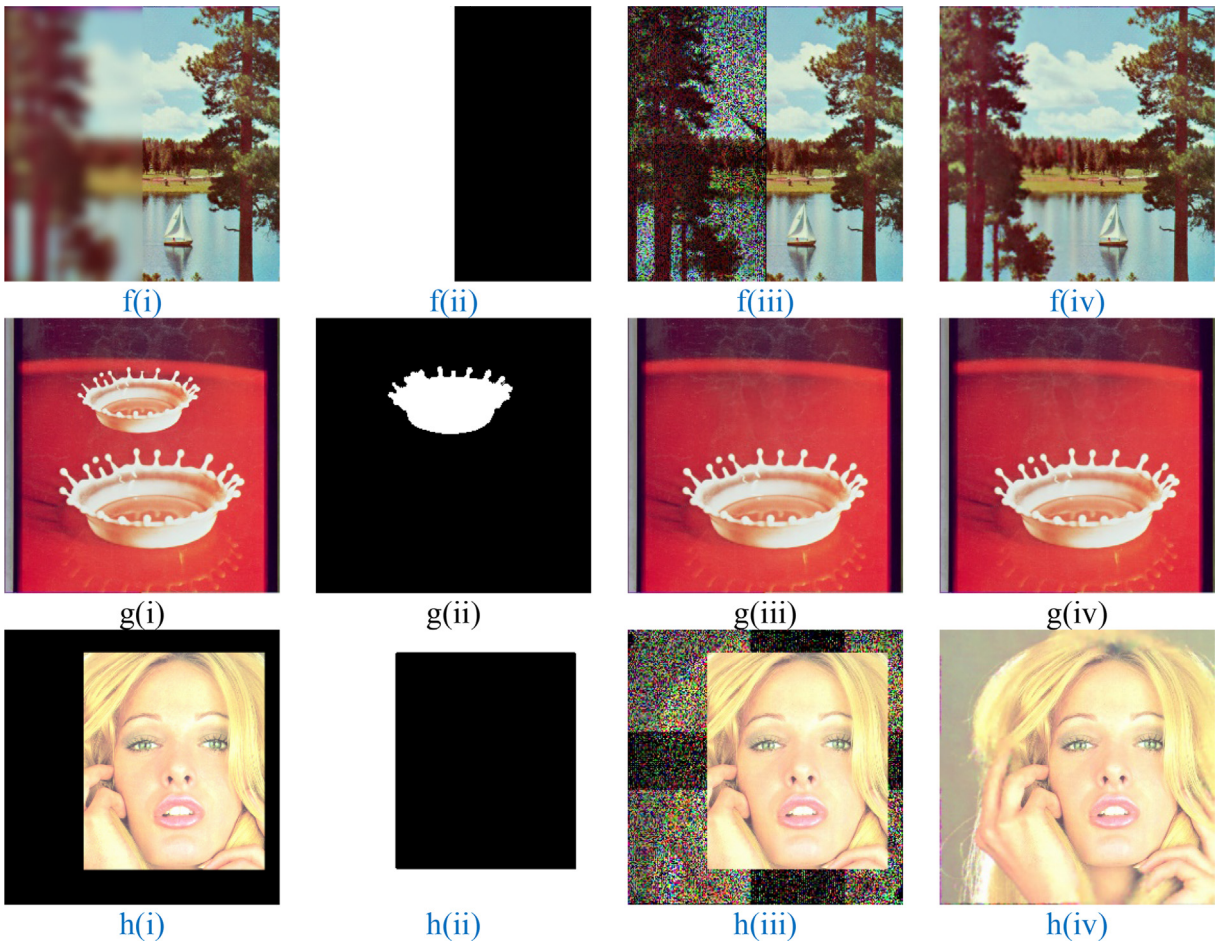

Fig. 7 (*continued*)

4.4. Recovered image quality

In the fourth set of experiments, the tampered images are recovered, and the results are then analyzed and compared based on the PSNR and SSIM. The PSNR and SSIM values are computed between the cover and recovered images. Thus, when no tampering is applied, the recovered image quality is identical to the watermarked image. When tampering is applied, the less the tampering rate, the higher the recovered image quality. In addition, a lower tampering rate means less tamper coincidence problem. Furthermore, The recovered image quality is compared in Tables 7 and Table 8.

With a 10% tampering rate, the Splash image has the highest recovered image quality since the splash image has less texture and edges than other test images. In contrast, the house image has the lowest quality, as it has a high density of textures and edges in the image. However, with an 80% tampering rate, the Baboon image has the most severe recovered image quality since the fur of the Baboon required a high density of textures and edges to be recovered. In addition, the highest difference in the PSNR value is between the 20% and 30% tampering rates, accounting for 3.47 dB. This is because the tamper coincidence problem starts to appear at a 25% tampering rate. However, the experiment on the 10% up to 40% tampering rates shows that the AuSR3 performs better than the existing schemes regarding the PSNR value. In addition, according to SSIM value, the AuSR3 outperforms the existing schemes on the 10% up to 50% tampering rates. The recovered image quality comparison between the AuSR3 and the existing schemes is shown in Tables 9–12.

The schemes by Tai (Tai and Liao, 2018) and Fan (Fan and Wang, 2018) did not successfully mitigate the tamper coincidence problem. Thus, it appears in the final recovered image and is amplified further using the random block mapping technique in their scheme. As a result, the recovered image quality was degraded significantly at a higher tampering rate. In comparison, the schemes by Molina-Garcia (Molina-Garcia et al., 2020) and Sinhal (Sinhal et al., 2020) mitigated the tamper coincidence problem using the image inpainting technique, which removes the artifacts of the tamper coincidence problem in the recovered image. However, they did not design the block mapping technique that prevents the tamper coincidence problem from occurring in the first place. Instead, they implemented the random block mapping technique to determine the embedding location of the recovery data. Therefore, the tamper coincidence problem began to appear at a small tampering rate of 10%, which degraded the recovered image quality.

4.5. Statistical analysis

The results of this study are analyzed using statistical analysis to determine the significance of the AuSR3 compared to the existing methods. At first, the characteristics of the data are analyzed using parametric analysis. It determines the normality and the distribution of the data. There are two possible outcomes of this analysis: parametric and non-parametric data. The paired *t*-test can be used for parametric data, while the Wilcoxon signed-rank test can be used when the data is non-parametric. The parametric analysis shows that the distribution of PSNR and SSIM is non-parametric.

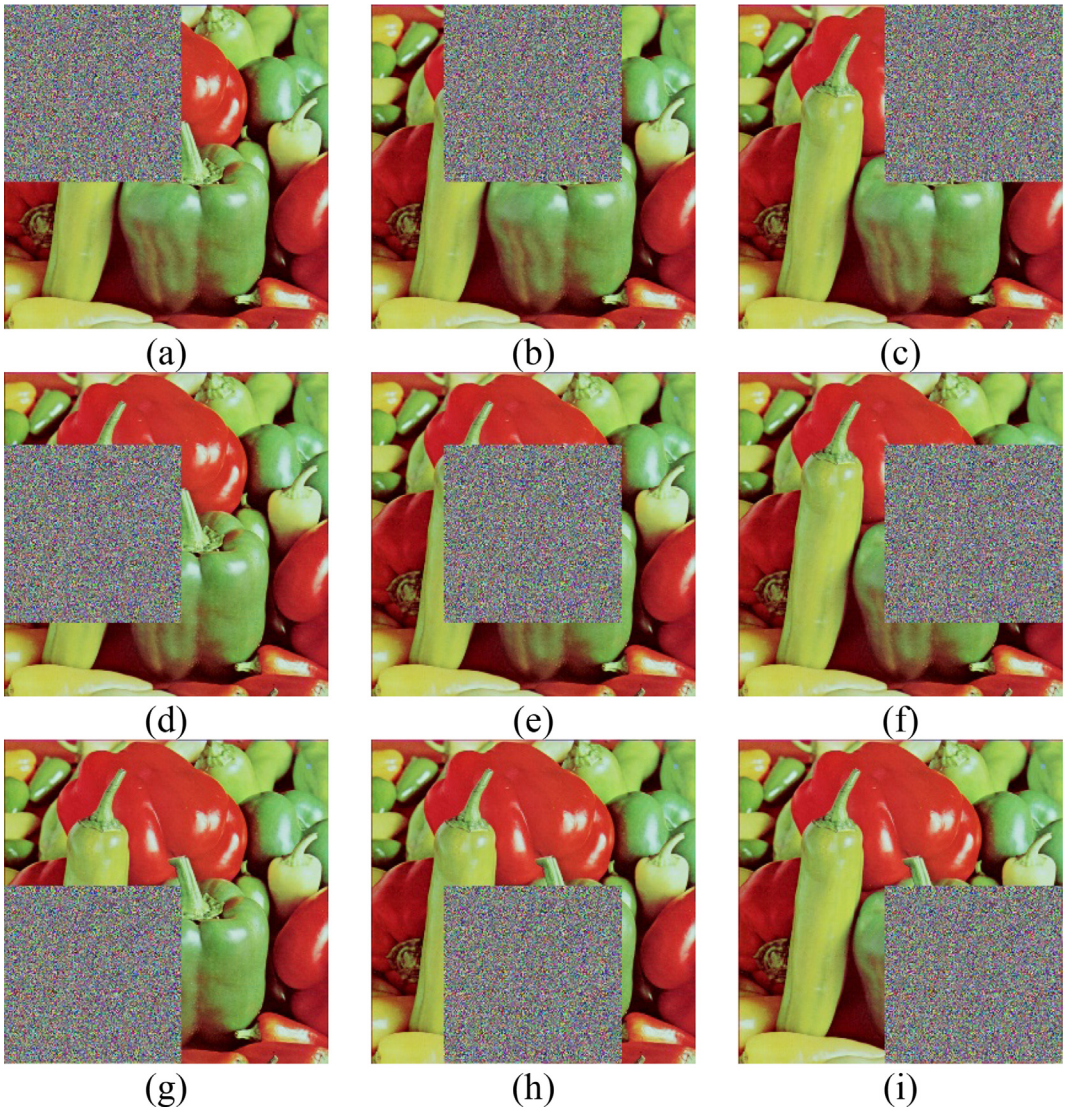

Fig. 8. Tampering rate of 30% in various locations on the Peppers image (a) Top left (b) Top center(c) Top right (d) Center left (e) Center (f) Center right (g) Bottom left (h) Bottom center (i) Bottom right.

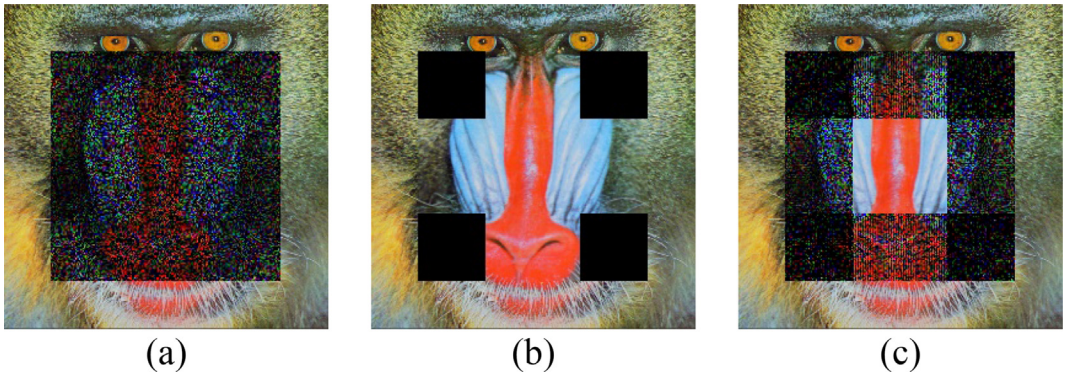

Fig. 9. The tamper coincidence problem with a tampering rate of 50% on the Baboon image(a) Random block mapping (b) Uniform block mapping (c) AuSR3 block mapping.

Therefore, Wilcoxon signed-rank tests are performed in this study. The test is conducted in pairs between two algorithms. The first test compares the AuSR3 to the AuSR1, while the second compares the AuSR3 and the AuSR2. Each evaluation and dataset are tested individually to provide robust information about the significance of the study. The statistical test of the watermarked image quality using the Wilcoxon signed-rank test is presented in Table 13.

In this statistical test, the null hypothesis H_0 refers to no significant improvement of the AuSR3. In contrast, the alternative hypothesis H_1 states that there is a statistically significant result

Table 6
The TCP rate under various tampering rates and locations on the USC-SIPI dataset.

Method	Tampering locations	Tampering rates							
		10	20	30	40	50	60	70	80
AuSR1 (Aminuddin and Ernawan, 2022)	Average	0.0186	0.0736	0.1555	0.2607	0.3775	0.5097	0.6409	0.7710
	Std. deviation	0.0013	0.0007	0.0014	0.0020	0.0022	0.0005	0.0029	0.0031
AuSR2 (Aminuddin and Ernawan, 2022)	Average	0.0131	0.0527	0.1178	0.2091	0.3282	0.4581	0.6115	0.7578
	Std. deviation	0.0002	0.0012	0.0014	0.0008	0.0007	0.0005	0.0008	0.0053
AuSR3	Top left	0.0000	0.0000	0.0445	0.1619	0.2973	0.4497	0.5887	0.7358
	Top center	0.0000	0.0000	0.0458	0.1666	0.3011	0.4518	0.5948	0.7359
	Top right	0.0000	0.0000	0.0443	0.1633	0.2987	0.4485	0.5935	0.7406
	Center left	0.0000	0.0000	0.0463	0.1675	0.3021	0.4542	0.5962	0.7363
	Center	0.0000	0.0000	0.0569	0.1814	0.3088	0.4571	0.6011	0.7359
	Center right	0.0000	0.0000	0.0479	0.1697	0.3051	0.4537	0.6003	0.7405
	Bottom left	0.0001	0.0001	0.0444	0.1627	0.2981	0.4485	0.5936	0.7407
	Bottom center	0.0000	0.0000	0.0471	0.1689	0.3050	0.4519	0.5990	0.7406
	Bottom right	0.0007	0.0009	0.0490	0.1686	0.3050	0.4496	0.5978	0.7451
	Average	0.0001	0.0001	0.0474	0.1678	0.3024	0.4517	0.5961	0.7390
	Std. deviation	0.0002	0.0003	0.0037	0.0055	0.0037	0.0028	0.0037	0.0031

Table 7
PSNR values comparison under various tampering rates on the USC-SIPI dataset.

TR	Airplane	Baboon	House	Lena	Peppers	Sailboat	Splash	Tiffany	Average
10	37.91	38.40	37.19	39.05	39.00	37.98	42.50	40.84	39.11
20	35.91	34.99	35.01	36.56	37.39	35.49	41.13	38.95	36.93
30	32.93	29.43	31.89	32.87	34.92	31.36	37.31	36.94	33.46
40	29.35	25.51	28.17	29.71	32.08	27.12	33.94	34.61	30.06
50	27.20	23.21	25.84	27.41	29.03	24.81	31.33	32.00	27.61
60	25.08	21.38	23.72	24.98	25.89	22.63	28.55	29.05	25.16
70	22.65	19.83	22.05	23.27	22.72	20.34	25.96	26.65	22.94
80	20.33	18.34	19.80	21.10	19.93	18.59	23.11	24.45	20.71

Table 8
SSIM values comparison under various tampering rates on the USC-SIPI dataset.

TR	Airplane	Baboon	House	Lena	Peppers	Sailboat	Splash	Tiffany	Average
10	0.9845	0.9966	0.9911	0.9969	0.9967	0.9946	0.9980	0.9971	0.9944
20	0.9794	0.9929	0.9853	0.9945	0.9953	0.9899	0.9975	0.9958	0.9913
30	0.9672	0.9625	0.9735	0.9881	0.9921	0.9761	0.9950	0.9939	0.9811
40	0.9395	0.9035	0.9456	0.9770	0.9867	0.9474	0.9907	0.9906	0.9601
50	0.9087	0.8415	0.9132	0.9641	0.9763	0.9158	0.9844	0.9837	0.9360
60	0.8659	0.7717	0.8709	0.9457	0.9572	0.8753	0.9738	0.9695	0.9038
70	0.8077	0.6948	0.8244	0.9286	0.9234	0.8191	0.9580	0.9522	0.8635
80	0.7417	0.6060	0.7576	0.9030	0.8818	0.7449	0.9310	0.9296	0.8120

Table 9
Recovered images comparison of the PSNR values on the USC-SIPI dataset.

TR	Tai (Tai and Liao, 2018)	Fan (Fan and Wang, 2018)	Molina-Garcia (Molina-Garcia et al., 2020)	Sinhal (Sinhal et al., 2020)	AuSR1 (Aminuddin and Ernawan, 2022)	AuSR2 (Aminuddin and Ernawan, 2022)	AuSR3
10	25.89	31.47	37.34	36.18	37.96	38.11	**39.11**
20	20.57	28.36	33.98	32.39	34.65	34.21	**36.93**
30	17.43	21.62	31.28	29.99	31.79	31.10	**33.46**
40	15.21	15.79	28.47	28.34	29.48	28.63	**30.06**
50	13.54	15.69	26.00	27.02	**27.64**	26.43	27.61
60	12.01	11.57	23.51	25.46	**25.72**	24.60	25.16
70	10.80	11.57	21.23	**23.95**	23.80	22.66	22.94
80	9.81	8.10	19.20	**22.47**	21.63	20.64	20.71

compared to the existing methods. The null hypothesis can be rejected when the p-value < α, where $\alpha = 0.05$, providing a 95% confidence level. In Tables 13 and Table 14, the +sign rejects the H_0, while the ∼sign can not reject the H_0. In addition, the Wilcoxon signed-rank test also provides a positive rank, which indicates the direction of change in the paired difference. When the positive rank is more significant than 50%, the AuSR3 performs better than the compared algorithm. According to Table 13, all the statistical test provides the p-value < α. It proves that the watermarked image quality of the AuSR3 is statistically significant compared to the previous method. Furthermore, most 1-to-1 comparisons show 100% positive ranks, which means that the AuSR3 performs better than the previous methods on all images within the selected dataset. The Wilcoxon signed-rank test is also performed to compare the recovered image quality, as presented in Table 14.

In terms of recovered image quality, the AuSR3 on the UCID-1338 dataset shows a significant improvement in the statistical test compared to other datasets with a p-value of 0.0. In addition,

Table 10
Recovered images comparison of the SSIM values on the USC-SIPI dataset.

TR	Tai (Tai and Liao, 2018)	Fan (Fan and Wang, 2018)	Molina-Garcia (Molina-Garcia et al., 2020)	Sinhal (Sinhal et al., 2020)	AuSR1 (Aminuddin and Ernawan, 2022)	AuSR2 (Aminuddin and Ernawan, 2022)	AuSR3
10	0.9384	0.9731	0.9714	0.9878	0.9928	0.9935	**0.9944**
20	0.8443	0.9502	0.9390	0.9768	0.9864	0.9864	**0.9913**
30	0.7364	0.8875	0.8977	0.9638	0.9742	0.9734	**0.9811**
40	0.6226	0.7230	0.8368	0.9504	0.9555	0.9534	**0.9601**
50	0.5135	0.7202	0.7571	0.9358	0.9339	0.9255	**0.9360**
60	0.3899	0.4249	0.6460	**0.9128**	0.9059	0.8932	0.9038
70	0.2744	0.4249	0.5157	**0.8843**	0.8705	0.8490	0.8635
80	0.1655	0.0094	0.3958	**0.8528**	0.8219	0.7937	0.8120

Table 11
Recovered images comparison on the Kodak-PCD0992 dataset.

TR	AuSR1 (Aminuddin and Ernawan, 2022)		AuSR2 (Aminuddin and Ernawan, 2022)		AuSR3	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
10	36.04	0.9841	36.74	0.9870	**37.05**	**0.9871**
20	32.53	0.9685	32.93	0.9735	**34.49**	**0.9783**
30	30.13	0.9482	30.23	0.9537	**31.79**	**0.9625**
40	28.28	0.9229	28.07	0.9267	**29.09**	**0.9344**
50	26.74	0.8932	26.20	0.8902	**27.06**	**0.9008**
60	**25.22**	0.8545	24.59	0.8461	25.16	**0.8577**
70	**23.76**	**0.8082**	23.07	0.7909	23.43	0.8073
80	**22.11**	**0.7493**	21.47	0.7245	21.65	0.7459

Table 12
Recovered images comparison on the UCID-1338 dataset.

TR	AuSR1 (Aminuddin and Ernawan, 2022)		AuSR2 (Aminuddin and Ernawan, 2022)		AuSR3	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
10	34.71	0.9803	35.26	0.9838	**35.81**	**0.9841**
20	30.93	0.9594	31.08	0.9646	**33.10**	**0.9726**
30	28.48	0.9331	28.27	0.9373	**30.27**	**0.9523**
40	26.47	0.8987	25.98	0.8993	**27.24**	**0.9143**
50	24.76	0.8578	24.05	0.8505	**24.95**	**0.8683**
60	**23.14**	0.8077	22.30	0.7894	22.93	**0.8123**
70	**21.44**	**0.7435**	20.59	0.7140	20.97	0.7424
80	**19.61**	**0.6645**	18.93	0.6279	19.07	0.6606

Table 13
Wilcoxon signed-rank test of the watermarked image quality.

Dataset	Evaluation	AuSR3 vs. AuSR1 (Aminuddin and Ernawan, 2022)			AuSR3 vs. AuSR2 (Aminuddin and Ernawan, 2022)		
		p-value	+rank	Sig.	*p*-value	+rank	Sig.
USC-SIPI	PSNR	0.0078	100%	+	0.0078	100%	+
	SSIM	0.0078	100%	+	0.0176	87.50%	+
Kodak-PCD0992	PSNR	1.19×10^{-7}	100%	+	1.19×10^{-7}	100%	+
	SSIM	1.19×10^{-7}	100%	+	1.19×10^{-7}	100%	+
UCID-1338	PSNR	2.58×10^{-220}	100%	+	2.58×10^{-220}	100%	+
	SSIM	1.54×10^{-218}	99.40%	+	1.65×10^{-214}	97.76%	+

Table 14
Wilcoxon signed-rank test of the recovered image quality.

Dataset	Evaluation	AuSR3 vs. AuSR1 (Aminuddin and Ernawan, 2022)			AuSR3 vs. AuSR2 (Aminuddin and Ernawan, 2022)		
		p-value	+rank	Sig.	*p*-value	+rank	Sig.
USC-SIPI	PSNR	0.0251	57.81%	+	2.11×10^{-10}	85.94%	+
	SSIM	0.3474	62.50%	∼	7.30×10^{-10}	93.75%	+
Kodak-PCD0992	PSNR	6.91×10^{-14}	69.79%	+	1.64×10^{-30}	90.63%	+
	SSIM	1.33×10^{-17}	80.21%	+	1.48×10^{-31}	94.79%	+
UCID-1338	PSNR	0.0	64.27%	+	0.0	91.72%	+
	SSIM	0.0	80.30%	+	0.0	94.37%	+

almost all the statistical tests can reject the H_0 hypothesis with p-value < α except for the SSIM values between the AuSR3 and AuSR1 on the USC-SIPI dataset. However, the positive rank on this dataset shows that 62.50% of the time, the AuSR3 produces higher SSIM values than the AuSR1. Therefore, The improvement of the AuSR3 is statistically significant compared to the previous methods.

4.6. Computational complexity

The computational complexity of the AuSR3 is computed based on the time it takes to execute the algorithm. The authentication and self-recovery scheme comprises three algorithms to be evaluated: watermark embedding, tamper detection and localization, and self-recovery. Several factors may influence the execution time, including the hardware environment, the software environment, the algorithmic complexity, the image size, and the tamper size. The computational complexity presented in this section is based on the same hardware and software environment for the entire experiment. The time complexity comparison of the watermark embedding across various datasets and algorithms is presented in Table 15.

The Kodak-PCD0992 dataset takes the longest to execute since it has the largest image size compared to the other datasets. In comparison, the UCID-1338 takes the least time to execute for being the smallest image size among the datasets. The algorithmic complexity also contributes to the execution time in the experiment. The proposed AuSR3 implements more complex block mapping, which adds the execution time compared to the existing schemes. Furthermore, the time complexity comparison of the tamper detection and localization is presented in Table 16.

The execution time for tamper detection and localization is negatively correlated to the size of the tampering area. This is because all of the algorithms presented in Table 16 implement the same three-layer authentication framework. The tamper localization emphasizes the detection of the non-tampered area of the image. Therefore, the larger the tampered region, the less time it takes to localize the image. At the same time, the image size positively correlates to the time complexity of the tamper detection and localization algorithm. In this comparison, the AuSR3 still requires the longest time to execute the tamper detection and localization. This is because the extraction process requires a block map to be reconstructed from the tampered image, adding more time complexity. Furthermore, the time complexity comparison of the self-recovery process is presented in Table 17.

Unlike tamper detection and localization, the self-recovery execution time is positively correlated to the tamper size of the image. The larger the tampered region, the longer it takes to recover the image. As mentioned earlier, the AuSR3 does not completely eliminate the tamper coincidence problem. It may still exist on the tampering rate larger than 25%. This problem is then solved using the image inpainting technique. However, the execution time is exponentially longer when the tampering rate exceeds 60%. This is because the tamper coincidence problem is densely populated in the four locations, as presented in Fig. 9(c). The denser the tamper coincidence problem, the more the image inpainting technique takes time to solve the affected block. Nevertheless, the AuSR3 still provides the highest recovered image quality compared to the existing techniques.

Table 15
Time complexity comparison of watermark embedding in seconds (s).

Dataset	USC-SIPI	Kodak-PCD0992	UCID-1338
AuSR1 (Aminuddin and Ernawan, 2022)	2.7150	3.8673	2.6424
AuSR2 (Aminuddin and Ernawan, 2022)	3.2750	4.7520	2.4046
AuSR3	3.4974	5.3985	2.7069

Table 16
Time complexity comparison of tamper detection and localization in seconds (s).

Method	Dataset	Tampering rates							
		10	20	30	40	50	60	70	80
AuSR1 (Aminuddin and Ernawan, 2022)	USC-SIPI	3.2160	3.1422	3.1060	3.0709	3.0325	2.9876	2.9877	2.9384
	Kodak-PCD0992	4.4299	4.3374	4.2686	4.2312	4.1385	4.0934	4.0276	3.9610
	UCID-1338	2.9787	2.9453	2.9051	2.8548	2.8350	2.8044	2.7685	2.7200
AuSR2 (Aminuddin and Ernawan, 2022)	USC-SIPI	2.3976	2.3404	2.3298	2.2947	2.2904	2.2627	2.2472	2.2301
	Kodak-PCD0992	3.5457	3.5106	3.4428	3.4444	3.4163	3.3950	3.3661	3.3277
	UCID-1338	2.0528	2.0352	2.0173	2.0079	2.0009	1.9850	1.9707	1.9521
AuSR3	USC-SIPI	3.4661	3.4061	3.3716	3.3444	3.3055	3.2750	3.2486	3.1766
	Kodak-PCD0992	5.2505	5.1782	5.1076	5.1065	5.0178	5.0063	4.9538	4.8315
	UCID-1338	2.6858	2.6497	2.6232	2.5944	2.5657	2.5372	2.5055	2.4789

Table 17
Time complexity comparison of self-recovery in seconds (s).

Method	Dataset	Tampering rates							
		10	20	30	40	50	60	70	80
AuSR1 (Aminuddin and Ernawan, 2022)	USC-SIPI	3.6347	3.8911	4.2469	4.8419	5.7555	7.6813	11.9470	24.2965
	Kodak-PCD0992	5.2963	5.6775	6.1937	7.0916	8.5967	11.5302	17.9704	35.9167
	UCID-1338	2.8005	2.9767	3.2658	3.7413	4.5177	5.9348	9.2553	18.4920
AuSR2 (Aminuddin and Ernawan, 2022)	USC-SIPI	4.0650	4.3667	4.8940	5.6775	7.0463	9.1633	13.6640	26.9634
	Kodak-PCD0992	6.0455	6.6334	7.6089	8.9741	11.1714	14.4635	21.2849	40.2820
	UCID-1338	3.1408	3.3864	3.7866	4.4179	5.3747	7.0185	10.6390	19.8766
AuSR3	USC-SIPI	4.1216	4.1485	4.4570	5.4356	7.2557	12.1951	26.0493	63.2257
	Kodak-PCD0992	6.2442	6.3323	6.7347	8.1435	10.8422	17.6329	35.1445	94.5124
	UCID-1338	3.4732	3.5372	3.7551	4.4905	6.0693	9.5647	19.3841	47.1154

5. Conclusion

This paper has presented a new block mapping to avoid the tamper coincidence problem in self-recovery. The new block mapping technique has embedded the recovery data into the most distant location possible, minimizing the tamper coincidence problem. It has eliminated the tamper coincidence problem by up to 25% tampering rates. When the tampering rate is higher than 25%, the tamper coincidence problem has been decreased by up to 10%. In addition, the improved LSB shifting algorithm has produced a high-quality watermarked image with 2 dB improvements in PSNR value compared to the LSB replacement technique. The new block mapping technique and improved LSB shiting algorithm have significantly contributed to the recovered image quality. A 20% tampering rate on all the images of the UCID-1338 dataset has been recovered by the AuSR3, resulting in the average PSNR and SSIM values of 33.10 dB and 0.9726, respectively. It has increased by up to 2 dB PSNR value compared to the existing technique. The Wilcoxon signed-rank test has been carried out to find the statistical significance of the AuSR3. Most of the test has shown that the AuSR3 has produced significant results in term of PSNR and SSIM compared to the previous methods.

The limitation of the proposed technique is that when the tampering rates are above 70%, the tamper coincidence problem is densely populated in four locations within the tampered region. While it can still be solved using the current image inpainting technique, executing the self-recovery process takes a high computational time on such high tampering rates. In future works, the image inpainting technique can be improved to incorporate a deep-learning technique such as super-resolution, which may reduce the computational time.

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.

Acknowledgements

This research is supported by the Postgraduate Research Grants Scheme (PGRS) of Universiti Malaysia Pahang Al-Sultan Abdullah under Grant UMP.05/26.10/03/PGRS220336 and Universitas Amikom Yogyakarta.

References

Aldahdooh, A., Masala, E., Van Wallendael, G., Barkowsky, M., 2018. Framework for reproducible objective video quality research with case study on PSNR implementations. Digit. Signal Process. 77, 195–206. https://doi.org/10.1016/J.DSP.2017.09.013.

Aminuddin, A., Ernawan, F., 2022. "AuSR1: Authentication and self-recovery using a new image inpainting technique with LSB shifting in fragile image watermarking. J. King Saud Univ. - Comput Information Scientist 34 (8B), 5822–5840. https://doi.org/10.1016/J.JKSUCI.2022.02.009.

Aminuddin, A., Ernawan, F., 2022. AuSR2: Image watermarking technique for authentication and self-recovery with image texture preservation. Comput. Electr. Eng. 102 (108207), 1–17. https://doi.org/10.1016/J.COMPELECENG.2022.108207.

Anand, A., Singh, A.K., 2020. Watermarking techniques for medical data authentication: a survey. Multimed. Tools Appl., 1–33 https://doi.org/10.1007/S11042-020-08801-0.

Anbu, T. et al., 2020. A comprehensive survey of detecting tampered images and localization of the tampered region. Multimed. Tools Appl. 80 (2), 2713–2751. https://doi.org/10.1007/S11042-020-09585-Z.

Armas Vega, E.A., González Fernández, E., Sandoval Orozco, A.L., García Villalba, L.J., 2020. Passive image forgery detection based on the demosaicing algorithm and JPEG compression. IEEE Access 8, 11815–11823. https://doi.org/10.1109/ACCESS.2020.2964516.

Bolourian Haghighi, B., Taherinia, A.H., Harati, A., 2018. 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, 49–64. https://doi.org/10.1016/J.JVCIR.2017.09.017.

Bolourian Haghighi, B., Taherinia, A.H., Mohajerzadeh, A.H., 2019. 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, 204–230. https://doi.org/10.1016/j.ins.2019.02.055.

Dadkhah, S., Abd Manaf, A., Hori, Y., Ella Hassanien, A., Sadeghi, S., 2014. An effective SVD-based image tampering detection and self-recovery using active watermarking. Signal Process. Image Commun. 29 (10), 1197–1210. https://doi.org/10.1016/J.IMAGE.2014.09.001.

Ernawan, F., Aminuddin, A., Nincarean, D., Razak, M.F.A., Firdaus, A., 2022. Three layer authentications with a spiral block mapping to prove authenticity in medical images. Int. J. Adv. Comput. Sci. Appl. 13 (4), 211–223. https://doi.org/10.14569/IJACSA.2022.0130425.

Ernawan, F. et al., 2022. Self-Recovery in Fragile Image Watermarking Using Integer Wavelet Transform. In: 2022 IEEE 8th Int. Conf. Smart Instrumentation, Meas. Appl., pp. 21–25, Nov., doi: 10.1109/ICSIMA55652.2022.9929127.

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

Huang, L., Xiang, Z., Li, J., Yao, H., Qin, C., 2022. New framework of self-embedding fragile watermarking based on reference sharing mechanism. Secur. Commun. Netw. 2022, 1–14. https://doi.org/10.1155/2022/2699802.

Hussan, M., Parah, S.A., Jan, A., Qureshi, G.J., 2022. Self-embedding framework for tamper detection and restoration of color images. Multimed. Tools Appl., 1–32 https://doi.org/10.1007/S11042-022-12545-4.

Kaur, M., Gupta, S., 2019. A fusion framework based on fuzzy integrals for passive-blind image tamper detection. Cluster Comput. 22 (5), 11363–11378 https://link.springer.com/article/10.1007/s10586-017-1393-3.

Liu, Y., Wang, H.H., Chen, Y., Wu, H., Wang, H.H., Jan. 2020. A passive forensic scheme for copy-move forgery based on superpixel segmentation and K-means clustering. Multimed. Tools Appl. 79 (1–2), 477–500 https://link.springer.com/article/10.1007/s11042-019-08044-8.

Molina-Garcia, J., Garcia-Salgado, B.P., Ponomaryov, V., Reyes-Reyes, R., Sadovnychiy, S., Cruz-Ramos, C., Feb. 2020. An effective fragile watermarking scheme for color image tampering detection and self-recovery. Signal Process. Image Commun. 81,. https://doi.org/10.1016/j.image.2019.115725 115725.

Mushtaq, S., Mir, A.H., 2014. Digital image forgeries and passive image authentication techniques: a survey. Int. J. Adv. Sci. Technol. 73, 15–32. https://doi.org/10.14257/ijast.2014.73.02.

Pourasad, Y., Ranjbarzadeh, R., Mardani, A., 2021. A new algorithm for digital image encryption based on chaos theory. Entropy 23 (3), 341. https://doi.org/10.3390/E23030341.

Prasad, S., Pal, A.K., 2020. Hamming code and logistic-map based pixel-level active forgery detection scheme using fragile watermarking. Multimed. Tools Appl. 79 (29–30), 20897–20928. https://doi.org/10.1007/s11042-020-08715-x.

Raj, N.R.N.N., Shreelekshmi, R., Feb. 2021. A survey on fragile watermarking based image authentication schemes. Multimed. Tools Appl. 80 (13), 19307–19333. https://doi.org/10.1007/S11042-021-10664-Y.

Ranjbarzadeh, R., Saadi, S.B., Amirabadi, A., 2020. LNPSS: SAR image despeckling based on local and non-local features using patch shape selection and edges linking. Measurement 164,. https://doi.org/10.1016/J.MEASUREMENT.2020.107989 107989.

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

Sahu, A.K. et al., 2023. A study on content tampering in multimedia watermarking. SN Comput. Sci. 4 (3), 1–11. https://doi.org/10.1007/S42979-022-01657-1.

Sahu, A.K., Sahu, M., Patro, P., Sahu, G., Nayak, S.R., 2023. Dual image-based reversible fragile watermarking scheme for tamper detection and localization. Pattern Anal. Appl. 26 (2), 571–590. https://doi.org/10.1007/S10044-022-01104-0.

Schaefer, G., Stich, M., 2004. UCID: an uncompressed color image database. Proc. Storage Retr. Methods Appl. Multimed. 5307, 472–480. https://doi.org/10.1117/12.525375.

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

Sinhal, R., Ansari, I.A., 2022. Tunable Q-Factor wavelet transform-based robust image watermarking scheme using logistic mapping and antlion optimization. Circuits Syst. Signal Process. 41 (11), 6370–6410. https://doi.org/10.1007/S00034-022-02090-8.

"SIPI Image Database." https://sipi.usc.edu/database/ (accessed Aug. 12, 2023).

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

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

"True Color Kodak Images." https://r0k.us/graphics/kodak/ (accessed Aug. 12, 2023).

Wei, W., Fan, X., Song, H., Wang, H., 2019. Video tamper detection based on multi-scale mutual information. Multimed. Tools Appl. 78 (19), 27109–27126. https://doi.org/10.1007/S11042-017-5083-1.