Multimedia Tools and Applications (2022) 81:18563–18594
https://doi.org/10.1007/s11042-022-12545-4

Self-embedding framework for tamper detection and restoration of color images

Muzamil Hussan[1] · Shabir A. Parah[1] · Aiman Jan[1] · G. J. Qureshi[2]

Received: 25 March 2021 / Revised: 20 January 2022 / Accepted: 31 January 2022 /
Published online: 9 March 2022

Abstract

The present era is paving huge expansion to the transmission of digital data in fields like health, military intelligence, scientific research, and publication media, etc. The nature of digital data makes it more vulnerable to various intentional and unintentional attacks and hence increases the necessity of verifying its integrity and authenticity. In this paper, a novel image watermarking method for tamper detection and recovery of color images has been proposed. The color image is primarily separated into three planes, and then each plane is divided into four equal halves. We sub-divide each half into 4 × 4 non-overlapping blocks. Further, from a group of four corresponding sub-blocks, a 32-bit watermark comprising of the arithmetic mean value along with the 8-bit data containing the location of the mapped block is generated. This 32-bit generated watermark is embedded into the two least significant bits (LSB) of the mapped block after being encrypted using gray code. The information is embedded in such a way using chaotic sequence such that a full restoration process can be carried out, even if 75% of blocks in any of the three planes get tampered with. This method obtains a high-quality restored image with an average PSNR value of 39.22 dB for the RGB model and 39.37 dB for the YCbCr model. The performance of the proposed method for various collage attacks shows its efficacy over other state-of-art techniques, making it a suitable candidate for the recovery of tampered color images.

Keywords Watermarking · Tamper detection · Tamper recovery · Image restoration

✉ Shabir A. Parah
shabireltr@gmail.com

[1] Department of Electronics and Instrumentation Technology, University of Kashmir, Srinagar, India

[2] Higher Education Department, J&K, Kashmir, India

1 Introduction

With the increase in image processing technologies and the availability of image editing software and tools, digital images can easily be manipulated and altered. To overcome this problem, authentication and tamper detection of digital images are essential in many fields like forensic investigation, surveillance systems, the entertainment industry, radiography imaging, etc. [3, 5, 20]. Tamper detection is classified into active and passive (blind) authentication methods. In the active authentication method, the information is generated from the image content itself and is used as a digital signature or digital watermark embedded inside the image. Whereas passive authentication methods do not need any digitally generated signature or embedded watermark to verify the authenticity or integrity of the digital images.

Authentication mechanism based on digital watermarking can be used for tamper detection and correction in digital images. Securing digital information is of paramount importance [1, 6, 38]. The security breaches result in tampering of impformation. Recovering the tampered area of an image is very important in many applications [16, 53]. Digital watermarking techniques can be categorized into robust and fragile watermarking. Robust watermarking methods withstand various signal processing attacks and show a slight effect on watermarked images [7, 9, 13, 50]. But a fragile watermark is highly affected when watermarked images are modified or attacked [17, 23, 29, 42]. For authentication purposes, fragile watermarking is typically used due to the ability to detect tamper in watermarked images. Fragile watermarking can be either pixel-based or block-based. In pixel-based methods, authentication depends on each pixel whereas block-based authentication depends on image blocks, as the authentication and recovery data is generated from the blocks of the image itself. Most of the recent authentication techniques are block-based and are employed to detect tamper while maintaining robustness against signal processing attacks like filtering and compression [14, 36, 39, 49]. These methods are mostly designed using robust watermarking in the transform domain like discrete cosine transform (DCT), discrete wavelet transform (DWT), etc. However, there is a conflict between watermarked quality and robustness which affects the imperceptibility of the watermark after embedding in the transform domain. Some of the watermarking techniques are designed to detect the tamper and recover the original data from various attacks like vector quantization attacks. A few such works can be seen in [2, 15, 25].

Many fragile watermarking techniques utilized for tamper detection, localization, and recovery could be seen in the literature [41, 46]. These techniques are robust against several attacks including VQ, collage attack, constraint average attack (CAA). In [34], a self-recovery three-level tamper detection technique has been proposed. In this scheme, the original digital image is divided into 4 × 4 blocks, and the generation of a watermark is done based on the average intensity of each block. The authentication method, however, is not able to detect the attacks like CAA, because it does not change the features of an image. The inability of the method is due to the less capacity of watermark embedding assigned for tamper detection and localization. Like in [31] a dual watermarking method has been proposed where two copies of watermarks are generated and are embedded in two different positions. In case the tamper is not detected by the first chance there is another chance for self-recovery. This scheme is vulnerable to various security attacks like collage attacks and constant average attacks. Sarreshtedari et al. [47], have presented a self-recovery scheme using Reed-Solomon coding. To retrieve the original image erasure location method (ELM) is used but this method can't recover the image data on n-k erasures. Coa et al. [8], have proposed a watermarking scheme using hierarchical-based image recovery. It uses the most significant bits, (MSB's) for image recovery which improve the chances of recovery. However, the method can be further enhanced by improving the perceptual quality of an image.

Keeping in view the drawbacks of the above-mentioned schemes, this paper aims to detect and recover the tampered blocks while providing a high degree of perceptual quality. Towards this end, we propose a new block-based fragile watermarking method using a chaotic algorithm for color images. To recover the maximum tampered regions in the color images multiple copies of the watermark are embedded into the four selected blocks in each plane of the (RGB, YCbCr) color image. The watermark generated from the blocks of the planes is used for authentication as well as for recovery purposes. In the proposed method even if 75% of the image gets tampered we are still able to recover the image with better visual quality.

1.1 Chaos theory

Chaos-based is the study of the unpredictable change or random behavior of a system. Chaotic maps are sensitive to initial conditions and are nonlinear in behavior. The behavior of chaotic maps is predetermined by mathematical formulae that make them deterministic. Chaos process is similar to stochastic processes as it occurs in nonlinear dynamical systems. Moreover, the nature of the stochastic process is non-periodic, non-convergent, and is particularly sensitive to initial circumstances. A chaotic system also resembles Gaussian noise because of its unpredictable nature [35]. Chaotic map-based techniques provide an excellent combination of speed, high security, complexity, appropriate computational overhead, and processing capability, they are viewed as attractive prospects for practical applications. The logistic map is considered one of the basic and simple methods in chaotic systems. It can be defined as

$$\text{ƌ}_{\boldsymbol{x+1}} = \beta\text{ƌ}_{\boldsymbol{x}}(\mathbf{1} - \text{ƌ}_{\boldsymbol{x}}) \tag{1}$$

Where x is the number of iterations in the chaotic sequence β and are the control parameters of the logistic mapping system that varies from 3.57 to 4 and 0 to 1 respectively. Because of the unpredictable and random behavior of the chaotic maps, we have used the basic logistic map for sub-block mapping to facilitate tamper detection and correction.

The rest of the paper is organized as follows. The literature review is presented in section 2. The proposed method is described in detail in section 3. Results and discussions are described in section 4. The paper concludes in section 5.

2 Literature review

Various watermarking techniques have been presented in the literature to protect the information contained in images. From the watermark embedding point of view, there are mainly two types of embedding, self-embedding and carrier image watermarking scheme. Self-embedding watermarking generate watermarks from the image features and embeds them into the image itself. In the carrier image watermark embedded is independent of the cover image in which it is embedded. The interpretation of the watermarked self-recovery schemes is usually expressed in terms of the quality of watermarked images with that of the restored images. The quality of the recovered image is generally measured by Peak Signal to Noise Ratio (PSNR), and Structural Similarity Index Metric (SSIM). The trade-off between these aspects is connected and needs to be correctly balanced. Restoration methods have gained the attention of researchers that mainly focuses on the quality and the tampering rate by checking the performance of the technique used to recover and authenticate the images [4, 21, 27, 28, 35, 54].

Lin et al. [33] proposed an authentication technique for digital images where the image is divided into 16 × 16 blocks. In this scheme, a discrete cosine transform (DCT) is applied to each block. The watermark generated from the blocks of the image is embedded in the DCT coefficients. This scheme shows 75% accuracy of still images under moderate compression and near or up to 90% under slight compression. Qian et al. [43] proposed a DCT based watermarking scheme to avoid tampering coincidence problem. In this scheme, the image is divided into 8 × 8 blocks, the DCT coefficients are encoded into a different number of bits. The authentication and restoration bits generated from the blocks are embedded into 3-LSB's of the original image. Because of the large block size in this method, tamper localization accuracy decreases. He et al. [18] proposed a self-recovery fragile watermarking scheme based on block-neighborhood tamper classification. The pseudorandom sequence is used to generate the non-linear block-mapping and implement an improved neighborhood characterization method for tamper detection. Patra et al. [40] proposed a fragile watermarking scheme for authentication and self-recovery of digital images. Chinese Remainder Theorem (CRT) is used in their scheme to provide a computational advantage with security measures. Their scheme fails to localize the tamper regions as the block size of the image is 8 × 8 which is large for localization. This scheme shows a PSNR of 36.77 dB for the recovered images with a small tamper ratio.

Huang et al. [19] proposed an image authentication fragile watermarking method for color images by using the concept of block truncation coding (BTC) for embedding. Their proposed scheme is vulnerable to attacks like collage and four scanning attacks, because of the blocking dependency of the algorithm. Tai et al. [51] proposed an effective image authentication watermarking scheme for tamper detection and self-recovery. This method is block-based and the watermark is generated using Integer Wavelet Transform (IWT) coefficients. A 3 × 3 block-neighborhood scheme is used for the recovery of the tampered regions of the image. Fan et al. [12] presented an improved version of the block-based scheme presented by Sarreshtedari and Mohammad [45], which is based on the Set Partitioning in Hierarchical Trees (SPIHT) to generate the watermark by dividing the image into 32 × 32 blocks. The main disadvantage of this scheme is its tamper coincidence problem that affects the recovered bits, which results in low imperceptibility during restoration. The works reported in [24, 44] are fragile-based watermarking methods, prone to higher tampering rates though they good quality in watermarked images than semi-fragile methods. These are gray-scale image techniques and use LSB substitution for embedding. References [32, 52] present block-based watermarking methods, where the most significant bits (MSB) of the blocks are used to generate the recovery watermarks. LSB-based embedding techniques in references [10, 51] embed a single type of image digest. In the methods presented in the references [11, 39, 48], watermarks generated for recovery are embedded only once. To generate the image digest the image is divided into 2 × 2 sub-blocks and DCT is applied on each block. Lastly, two bits of the coefficients of each block are extracted. In [37], an effective fragile watermarking technique for tamper detection and recovery of color images has been presented. In this scheme, recovery watermarks are generated using the YCbCr planes of the color image, where three copies of the recovery watermarks are embedded to help in the recovery of the image quality after high tampering rates. With the increase in tampering rate, this scheme shows less perceptibility with an average PSNR of 37.34 dB SSIM of average SSIM of 0.9714. The schemes present in literature either lack image quality or lack ability to recover the tampered region. To overcome these limitations a scheme should be designed to build up the visual quality of an image while enhancing the recovery capability.

In the proposed approach, a self-embedding watermarking technique for tamper detection and correction for the color images has been proposed which is the extension of paper [22]. In the proposed method the color image is separated into three plane components (RGB, YCbCr). Each plane of the color image is distributed into four equal blocks, and then these blocks are sub-divided into 4 × 4 blocks. The watermarks are generated from the four corresponding 4 × 4 sub-blocks of the plane and are embedded according to the mapping sequence generated by a chaotic algorithm for each plane respectively. The embedding capacity of the proposed approach is 2 bpp. The proposed scheme not only shows good imperceptibility but also better restoration proficiency which enhances the reliability, legitimacy and security of this method. Table 1 shows the comparative analysis of the related work.

3 Proposed work

The block diagram of the proposed method is shown in Fig. 1. In the proposed scheme the color image is separated into three planes and each plane is divided into four equal blocks, then these four blocks of all the three planes are sub-divided into 4 × 4 non-overlapping sub-blocks. It is a self-embedding scheme where the watermark is generated from the blocks themselves and is embedded in the planes of the color image using the chaotic sequence generated by using chaotic mapping. The proposed method is used to detect tamper as well as to restore the tampered regions (Fig. 2).

3.1 Block mapping

In the proposed scheme chaos is used to generate the mapping sequence. Each 4 × 4 block of the plane in the sequence of the color image is represented by a distinctive symbol. For authentication and recovery purposes the mapping block of the plane is set to be complementary to those blocks where watermark/information of the blocks has been attained from the plane. Watermark embedding is done according to the mapping sequence of a plane generated and is equal to the number of blocks in a plane using chaos. If the mapping address and actual address of the plane are similar to any point, the adjustment is done accordingly so for dissimilar address block we get, dissimilar mapping block. Here in the proposed method, the original color image is separated into three planes. Each plane is separated into four equal blocks i.e. I_q where I is equal to the plane number that runs from 1:3 and ‘q’ is a block number that runs from 1:4. Then all planes are further sub-divided into 4 × 4 sub-blocks represented by I_q, b where ‘b’ runs from l/2 × m/2 × 1/16. To generate the block sequence key, an initial value of β =3.9 and = 0.5 is used in eq. (1) and the number of iterations is equal to the number of blocks of an image/block size. As can be seen in Fig. 3 the location/position of the blocks gets changed using chaos. In the figure, the color image is first separated into three planes then the first plane is divided into four equal blocks. These blocks are further divided

Table 1 Comparative analysis of the related work

Techniques	Domain used	Average PSNR dB	Methodology/Advantages	Limitations
Bhalerao [5]	Spatial domain	51.12	Hash has been used for tamper detection	The number of false-positive pixels were high
Ansari [3]	Spatial Domain	44.07	Fragile watermarking technique for tamper detection, localization, and recovery, Embedding in 2lsb's.	The number of false-positive pixels is high.
Chamlawi [9]	DWT domain	–	Tamper detection, and recovery capability	Low self-recovery rate
Qi [42]	DWT domain	41.39	Robust to unintentional attacks and fragile to intentional attacks	High computational complexity, poor localization accuracy
Singh [39]	DCT domain	37.41	Tamper detection, localization, and recovery capability	High computational complexity
Gull [15]	Spatial domain	51	Tamper detection and localization capability, Less computational complexity	Cannot withstand unintentional attacks and cannot recover tampered regions
Azeroual [4]	Spatial domain	51	Fragile watermarking technique for tamper detection, localization	Shows less perceptual quality
Lin [33]	DCT domain	37	Resist noise or compression attacks	This scheme shows 75% accuracy of still images under moderate compression and near or up to 90% under slight compression
Qian [43]	DCT domain	44.01	High tamper detection capability	Large block size results in decreased, tamper localization accuracy.
He [18]	Spatial domain	36.77	Computationally efficient due to usage of modular arithmetic	Fails to localize the tampered regions
Tai [51]	IWT domain	44.08	Authentication and recovery bits are generated using IWT transformation, and the embedding is based on LSB	Tampering coincidence problem.
Fan [12]	SPIHT domain	44.07	The algorithm integrates recovery data bits to attain better robustness against the tampering coincidence problem	Poor objective quality of the reconstructed image.
Saeed [45]	Spatial domain	44.15	SPIHT is used for compression.	Tamper coincidence problem resulting in low imperceptibility during restoration.
Rajput [44]	Spatial domain	44.07	Shows high fragility	High False Positive pixels
Tong [52]	Spatial domain	40.70	Tamper detection and recovery capability using chaotic map	Shows less perceptual quality after recovery of tampered blocks.
Molina [37]	Spatial domain	44.63	Tamper detection and self-recovery capability	Shows less imperceptibility.

into 4 × 4 sub-blocks and each 4 × 4 block is represented by a single cell signified by numbers. After applying the chaotic algorithm the position of the blocks gets changed like C1 is mapped with C11 and so on. The same process is applied to the other two planes as well.

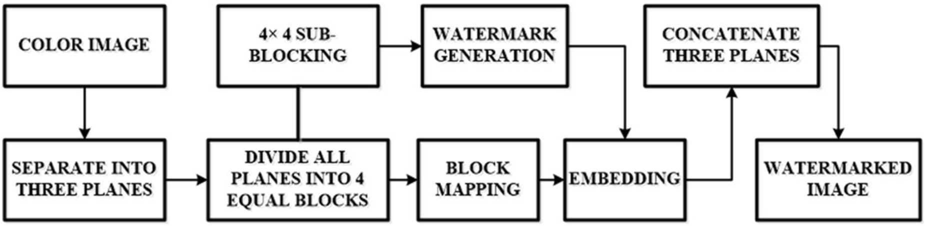


Fig. 1 Block diagram for the proposed method

3.2 Watermark generation

In the proposed scheme the color cover image is separated into three plane components (i.e. RGB, YCbCr). Then each plane of the color image M of size l × m is divided into four equal blocks Bx, where B is equal to plane number runs from B = 1:3, x is block number of the plane runs from x = 1:4, and the size of each block is l/2 × m/2. Further, the four blocks of all the three planes are sub-divided into 4 × 4 sub-blocks i.e. Bxy (where B = 1:3, x = 1:4, y = l/2 × m/2 × 1/16). The watermark is generated after calculating the arithmetic mean (average intensity) of all four corresponding sub-blocks of the plane after setting the two LSB's of each pixel equal to zero. Figure 4 illustrates the generation of watermarks for clear understanding.

The average intensity of only six bits is taken after the right shifting of two average bits of each block. Then these six-average bits from all four blocks are concatenated to form a 24-bit watermark which is then joined with the location of the mapping block. Thus, we obtain a 32-bit watermark that is embedded in the LSB's of the corresponding mapping four sub-blocks of the plane. In the embedding process, the mere difference is the eight-bit location data which is different for a different block. Further, these 24-bits average intensity with 8-bits location is repositioned in such a way that the first six bits from average intensity bits are followed by the first two bits of location bits and so on. After reorganizing the 32-bit watermark binary to gray code encryption is performed to enhance the security of the watermark. This encrypted watermark is then embedded in the equivalent sub-blocks of the mapping blocks. The technique is performed on all the blocks of the plane. The whole process is done to all the three planes of the color image and then concatenating all the three planes to get a watermarked image. The Pseudocode for watermark generation and data embedding is presented in Algorithm 1.

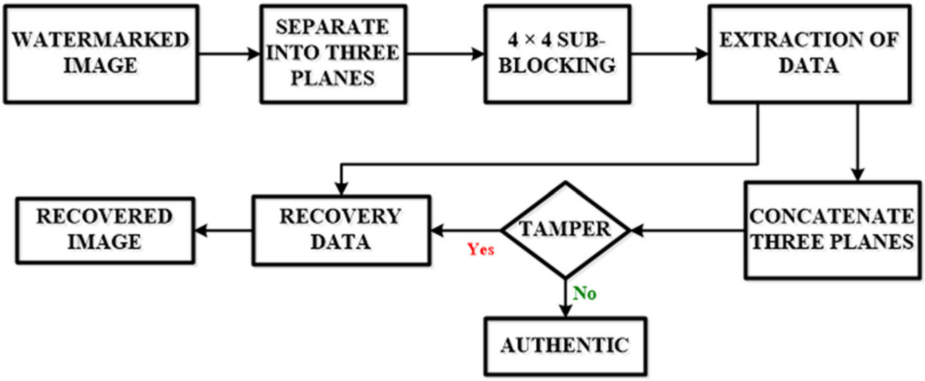


Fig. 2 Extraction process of the proposed method

3.3 Embedding

In the proposed approach, first, the color image is separated into three planes i.e. RGB and YCbCr. Then each plane of the color image is divided into four equal blocks. All the four blocks of all the three planes are further sub-divided into 4 × 4 sub-blocks. Then the watermark generated as discussed in the watermark generation process is embedded in the 2 LSB's of each pixel in the 4 × 4 mapping blocks of the corresponding plane. The same embedding procedure is followed throughout the various locations in all three planes of the original color image to get the watermarked color image as depicted in Algorithm. Pertinent to mention the *mod (T, 255)* represents modulous function used to locate address of the image blocks utilizing various image dimentions.

Algorithm 1: Pseudocode for watermark generation and data embedding.

```
Input: Color image ‘M (l, m)’, β, Ƌ x+1., La location address
Output: Watermarked image ‘WI (l, m)’.

BEGIN
M ⟵ M1, M2, M3
M ⟵ 1:3
M1 ⟵ First plane
S ⟵ l/2×m/2
T ⟵ l/2×m/2×1/16
α ⟵ avrg (T)
for rounds ⟵ 1:T – 1
        $Ƌ_{x+1} = β Ƌ_x(1 - Ƌ_x)$
    end for
La = mod(T,255)
Wm ⟵ α +La
Wm = Rearrange(WM1,WM2,WM3)
Encrypt(WM1,WM2,WM3)
WI ⟵ Cat(WM1,WM2,WM3)
END
```

3.4 Tamper detection and recovery

In this section, we discuss the tamper detection and recovery process adopted in our method. The Pseudocode is represented in Algorithm 2.

Algorithm 2: Pseudocode for data extraction.

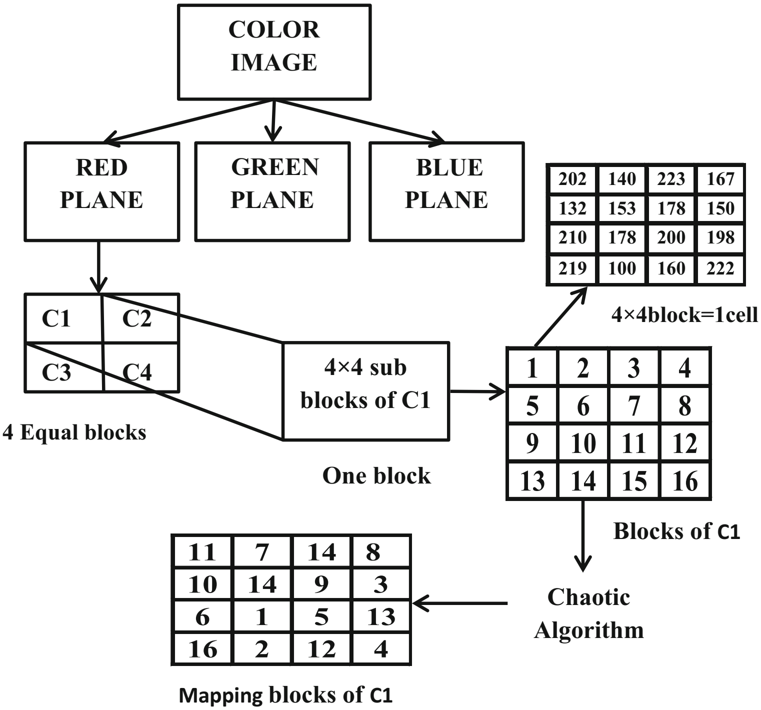
```
Input: watermarked image ‘$WI_{(l, m)}$’, β, $ƌ*_{x+1}$, La location address
Output: Recovered image ‘$RI_{(l, m)}$’.

BEGIN
M* ← M1*, M2*, M3*
M* ← 1: 3
M1* ← First plane
S* ← l /2×m/2
T* ← l/2×m/2×1/16
α* ← avrg (T*)
for rounds ← 1: T* – 1
        $ƌ*_{x+1} = β ƌ_x (1 – ƌ_x)$
    end for
La* = mod(T*,255)
Wm* ← α* - La*
Wm* = Rearrange(WM1*,WM2*,WM3*)
Decrypt(WM1*,WM2*,WM3*)
if WI ← cat(WM1*,WM2*, WM3*)   then authentic
else  unauthentic
end if
ENDs
```

To detect whether the image has been tampered with or not, the received watermarked color image is separated into three plane components (i.e. RGB, YCbCr). Then each plane of the watermarked color image WI of size l × m is separated into four equal blocks B’x (where B′ is equal to plane number runs from 1:3, x = 1:4 and size of each block is l/2 × m/2). Further, the blocks of all the three planes are sub-divided into 4 × 4 sub-blocks i.e. B’xy (where B′ = 1:3, x = 1:4, y = l/2 × m/2 × 1/16). Retrieve the 32-bit watermark from the 2 LSB’s of each pixel of all the four corresponding mapping blocks of the plane. Decrypt the watermark from gray to binary and reshuffle in the opposite way as in the generation of the watermark process. Compare the retrieved data with that of the embedded data, if both are the same then it is authentic otherwise tampered with. Repeat the procedure for all three planes of the color image for tamper detection. Using binary-to-gray encryption makes the watermark more sensitive towards any slight change. If obtained data is not equal to extracted data, the block is considered invalid in the plane and is recovered using the recovery process.

3.5 Recovery

The blocks in the color image planes that are marked tampered or invalid need to be recovered. Those tampered blocks of the planes are recovered by obtaining the watermark and location of the other three corresponding blocks of each plane. From each parent block, a 32-bit watermark is retrieved from the two LSB’s from each pixel of the block of the plane and is



202	140	223	167
132	153	178	150
210	178	200	198
219	100	160	222

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

11	7	14	8
10	14	9	3
6	1	5	13
16	2	12	4

Fig. 3 Example of mapping

decrypted from gray code encryption. Then rearranging all these bits in the same position as arranged them before embedding them in the mapping blocks. Separate the intensity value and location value bits to reconstruct the blocks.

4 Experimental results

In this section, various test color images (general as well as medical) of size 512 × 512 shown in Fig. 5 have been used to evaluate the performance of the proposed method. In the proposed method 2 LSB's of the pixels of all the three planes of the color images are used to embed the watermark which is generated from the blocks of the image planes. The experimental simulation has been performed on Intel (R) Celeron(R) CPU N3350 (1.10GHz) with 4 GB RAM using MATLAB 2016a on windows 10.

The performance of the proposed algorithm has been evaluated in terms of its ability for tamper detection, localization, and recovery of the images. Various objective quality metrics like Peak Signal to Noise ratio (PSNR), and Structural Similarity Index (SSIM) have been calculated for the proposed scheme. Table 2 shows the subjective quality of our scheme when various watermarked images are subjected to attacks like text addition, copy and paste attack, and crop or content removal attack respectively. Experimental evaluation has shown that our algorithm is efficient enough to detect tamper. Further, the recovered image of high perceptual quality is itself testimony that the proposed technique is effective in the recovery of tampered regions.

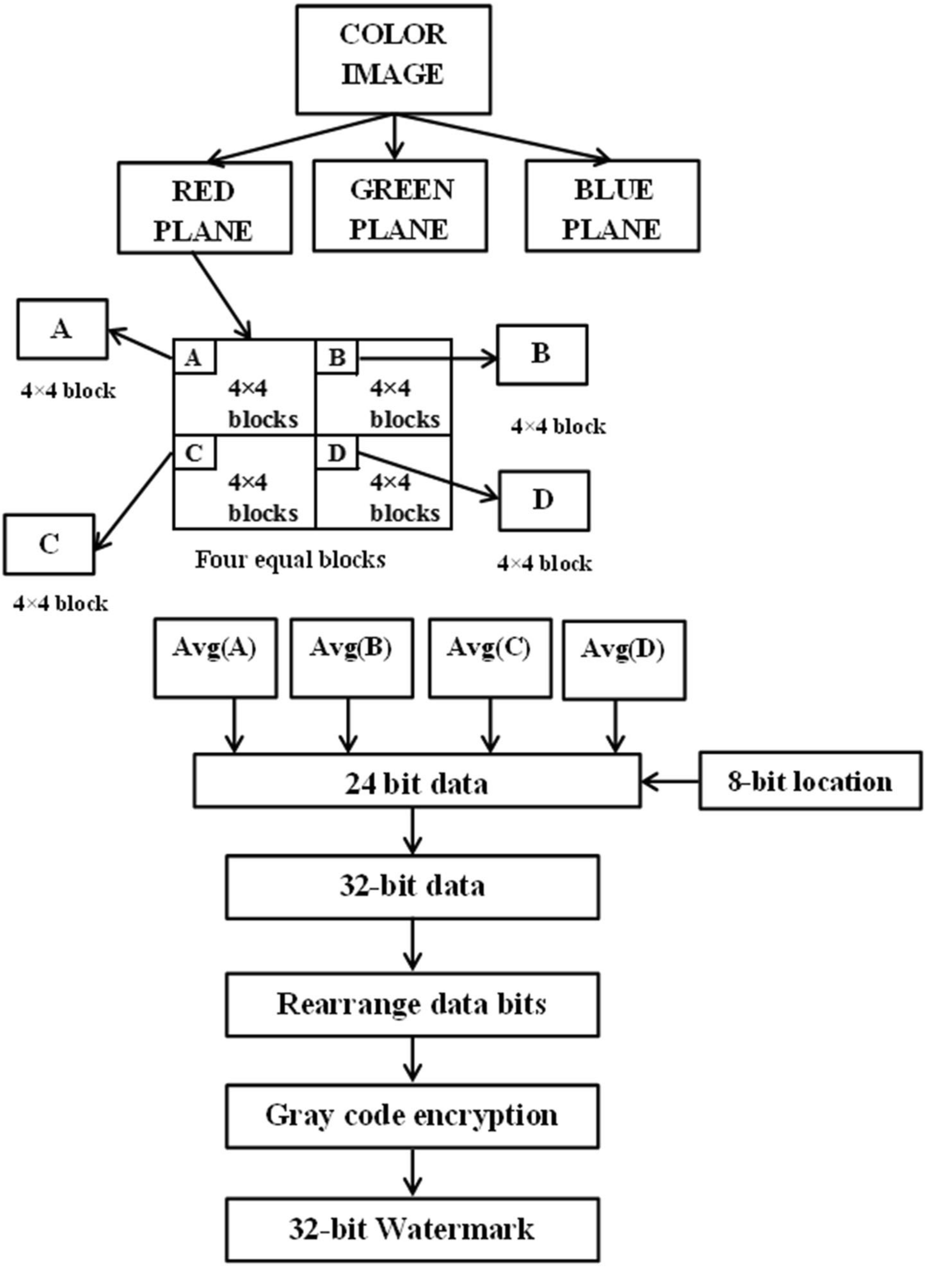


Fig. 4 Watermark generation

4.1 Perceptual quality assessment

To check the quality of the watermarked image compared to the original image, PSNR and SSIM are calculated between the original and watermarked image. Let the original color image

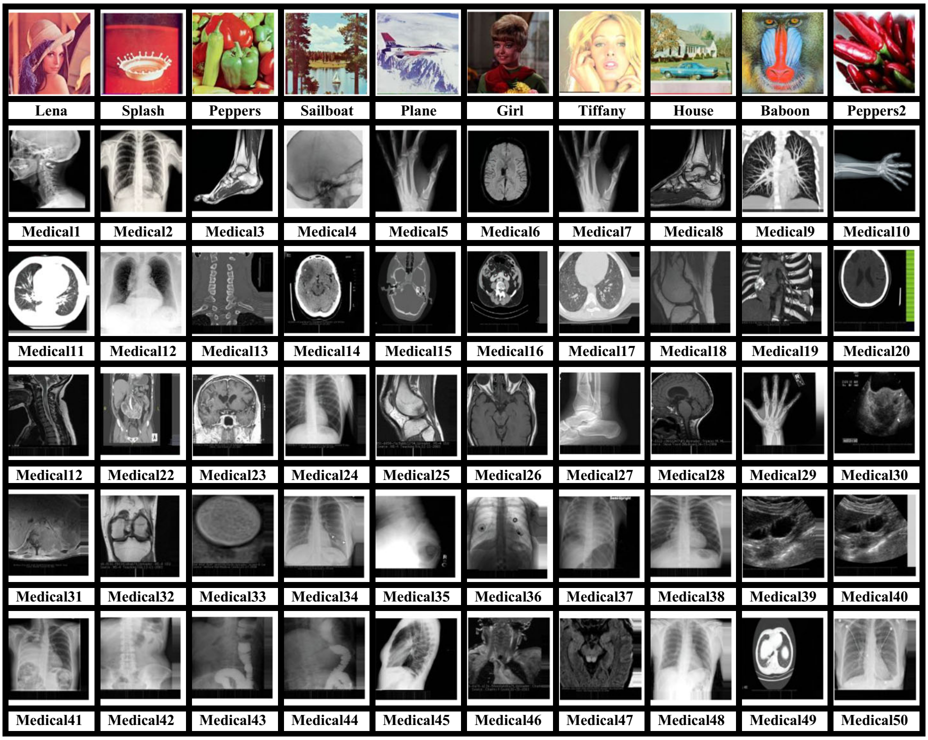


Fig. 5 Test images used in the proposed method

is represented by 'M' and the watermarked color image is represented by 'N', then PSNR is calculated as:

$$\mathrm{PSNR} = 10\log\frac{(2^{\mathrm{n}}-1)^2}{\mathrm{MSE}} = 10\log\frac{(255)^2}{\mathrm{MSE}}\,dB \tag{2}$$

Where n is the minimum number of bits in a given image that can represent the maximum possible intensity. MSE is the mean square error that can be calculated as follows:

Table 2 Imperceptibility analysis of the proposed scheme on general images

Images	PSNR (RGB)	SSIM	PSNR (YCbCr)	SSIM
Lena	44.98	0.9989	44.99	0.9981
Milk drop	44.84	0.9975	44.89	0.9988
Peppers	44.76	0.9988	44.87	0.9980
Sailboat	44.87	0.9968	44.83	0.9983
Plane	44.70	0.9830	44.97	0.9979
Girl	44.64	0.9881	44.72	0.9970
Baboon	44.92	0.9984	44.95	0.9970
House	44.70	0.9950	44.83	0.9987
Tiffany	44.89	0.9978	44.91	0.9974
Peppers2	44.12	0.9922	44.14	0.9981

$$\mathrm{MSE} = \frac{1}{\mathrm{PQ}} \sum_{i=1}^{P} \sum_{j=1}^{Q} \left(\mathrm{M}_{i,j} - \mathrm{N}_{i,j}\right)^2 \tag{3}$$

Here P and Q are the dimensions of the image.

SSIM of the original color image and a watermarked color image is calculated as

$$\mathrm{SSIM} = \frac{(2\mu_a\mu_v + e1)(2\sigma_{av} + e2)}{(\mu_a{}^2 + \mu_v{}^2 + e1)(\sigma_a{}^2 + \sigma_v{}^2 + e2)} \tag{4}$$

Where μa, μv are local means, σa, σv are standard deviations and σav is cross-covariance of two-images M and N and e1, e2 are the constants. Tables 2 and 3 show the imperceptibility analysis of the watermarked color images (general as well as medical) of the proposed method while Tables 4 and 5, shows the comparison of the watermarked color images of the proposed scheme with other existing techniques.

Tables 2 and 3 show the subjective analysis of the proposed scheme after embedding the watermarks on both the color image planes (RGB, YCbCr) of the general as well as medical images. The comparison results in terms of PSNR and SSIM of the watermarked color images of the proposed method with other techniques are presented in Tables 4 and 5 separately. A comparison of our method with state-of-the-art shows that our method provides better results in terms of visual quality.

4.2 Computational complexity analysis

For high-speed and real-time applications, the computational efficiency of a watermarking system is essential. The amount of time it takes to embed and extract information determines the efficiency of the watermarking method. The system is said to be computationally efficient if the processing time of the algorithm is less. Figure 6 shows the comparison of the embedding time between the proposed method and the approach presented in [49]. The computational time of the proposed watermarking methodology is low and thus is better than the scheme reported in [49].

4.3 Tamper detection and recovery ability

The proposed procedure has been tested for its capability for tamper detection and recovery by subjecting the watermarked color images to various attacks including copy and move attack, content removal attack, and text addition attack. The results obtained after the subjection of watermarked images to these attacks are present in Tables 6 and 7 depicts the Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index (SSIM) of the proposed scheme. Figures 7 and 8 show the subjective quality of the watermarked color images (general as well as medical) of the proposed approach after image attacks (i.e. copy and move attack, content removal attack, and text addition attack). The scheme shows its ability to detect as well as to recover the tamper if any present in the watermarked image. Further, the recovery process of the proposed method provides a recovered image with better perceptual quality.

Tables 6 and 7 show the subjective analysis of the watermarked images (general, medical) of both the planes (RGB, YCbCr) of the proposed method after subjected to various kinds of attacks like crop, text addition, and copy-paste. It can be seen from the results that the proposed

Table 3 Imperceptibility analysis of the proposed scheme on medical images

Images	PSNR (RGB)	SSIM	PSNR (YCbCr)	SSIM
Medical1	43.90	0.9254	43.84	0.9969
Medical2	43.98	0.9541	43.96	0.9974
Medical3	44.51	0.9138	43.79	0.9990
Medical4	44.17	0.9706	44.01	0.9901
Medical5	44.09	0.9106	43.79	0.9979
Medical6	44.50	0.9141	43.77	0.9957
Medical7	44.10	0.9106	43.79	0.9979
Medical8	44.39	0.9631	44.14	0.9979
Medical9	44.17	0.9841	43.99	0.9981
Medical10	44.90	0.9024	43.57	0.9983
Medical11	42.41	0.9783	43.72	0.9987
Medical12	44.18	0.9741	43.99	0.9965
Medical13	44.18	0.9662	43.98	0.9972
Medical14	44.27	0.9406	43.89	0.9970
Medical15	44.63	0.9440	43.94	0.9982
Medical16	44.49	0.9095	43.74	0.9988
Medical17	44.19	0.9808	44.02	0.9965
Medical18	44.16	0.9729	43.96	0.9958
Medical19	44.16	0.9816	43.99	0.9982
Medical20	44.06	0.9121	43.79	0.9993
Medical21	44.08	0.9717	43.95	0.9986
Medical22	43.72	0.9361	43.81	0.9957
Medical23	44.19	0.9852	44.00	0.9947
Medical24	44.12	0.9673	43.98	0.9958
Medical25	44.14	0.9812	43.98	0.9981
Medical26	44.14	0.9771	44.02	0.9980
Medical27	43.79	0.9766	43.91	0.9958
Medical28	44.18	0.9525	43.94	0.9989
Medical29	43.44	0.9265	43.91	0.9983
Medical30	45.15	0.9774	44.23	0.9989
Medical31	44.16	0.9761	43.99	0.9958
Medical32	43.99	0.9793	43.95	0.9963
Medical33	44.13	0.9676	43.98	0.9967
Medical34	44.14	0.9740	44.00	0.9935
Medical35	43.71	0.9392	43.84	0.9952
Medical36	44.26	0.9772	44.03	0.9953
Medical37	44.41	0.9700	44.07	0.9941
Medical38	44.15	0.9737	44.01	0.9930
Medical39	44.05	0.9541	43.91	0.9979
Medical40	44.04	0.9385	43.86	0.9980
Medical41	44.42	0.9750	44.09	0.9933
Medical42	44.12	0.9727	44.01	0.9928
Medical43	44.13	0.9683	43.99	0.9954
Medical44	44.13	0.9680	43.99	0.9947
Medical45	44.98	0.9747	44.23	0.9972
Medical46	44.23	0.9779	44.01	0.9978
Medical47	44.20	0.9779	44.02	0.9986
Medical48	44.16	0.9725	44.00	0.950
Medical49	44.52	0.9220	43.84	0.9991
Medical50	44.15	0.9713	43.98	0.9929

scheme successfully restores the tampered region/portion of the color images even if subjected to a variety of tests as discussed.

Table 4 Comparison of PSNR (dB) for watermarked images

	Images								
	Lena	Sailboat	Peppers	Baboon	House	Splash	Tiffany	F-16	Average
Singh[49]	37.90	37.90	37.79	37.90	37.88	37.84	37.44	37.88	37.81
Dadkhan[11]	44.13	44.12	44.06	44.14	44.19	44.08	43.85	44.12	44.08
Tong[52]	37.90	37.90	37.79	37.90	37.88	37.84	37.44	37.88	37.81
Fan[12]	44.13	44.10	44.06	44.12	44.18	44.08	43.84	44.11	44.07
Tai[51]	44.12	44.11	44.06	44.14	44.14	44.09	43.85	44.12	44.08
Molina [37]	44.60	44.61	44.54	44.64	44.66	44.47	44.87	44.69	44.63
Proposed	44.98	44.87	44.76	44.92	44.70	44.84	44.89	44.70	44.83

Table 5 Comparison of SSIM for watermarked images

	Images								
	Lena	Sailboat	Peppers	Baboon	House	Splash	Tiffany	F-16	Average
Singh[49]	0.9307	0.9493	0.9234	0.9763	0.9319	0.8942	0.9246	0.9194	0.9312
Dadkhan[11]	0.9820	0.9868	0.9791	0.9941	0.9815	0.9695	0.9806	0.9782	0.9814
Tong[52]	0.9307	0.9494	0.9234	0.9763	0.9319	0.8942	0.9246	0.9194	0.9312
Fan[12]	0.9820	0.9867	0.9791	0.9941	0.9815	0.9695	0.9804	0.9781	0.9814
Tai[51]	0.9820	0.9868	0.9791	0.9941	0.9815	0.9695	0.9805	0.9781	0.9814
Molina [37]	0.9840	0.9884	0.9816	0.9947	0.9834	0.9737	0.9846	0.9812	0.9839
Proposed	0.9989	0.9968	0.9988	0.9984	0.9950	0.9975	0.9978	0.9830	0.9957

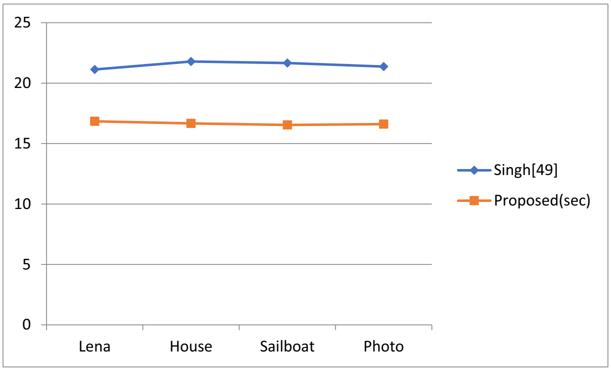


Fig. 6 comparison of embedding time (seconds)

4.4 Self-recovery evaluation under different tampering rates

The algorithms designed for tamper detection with self-recovery competencies can localize the tampered areas and can recover the damaged part of the color image. In different tampering rate attacks, a watermarked image is attacked with different percentage rates. In this type of attack, we have used 512 × 512 sized color images (RGB & YCbCr) of general as well as

Table 6 Subjective quality of the proposed method on different planes of the restored color general images

Images	Attacks											
	Crop				Text addition				Copy Paste			
	RGB plane		YCbCr plane		RGB plane		YCbCr plane		RGB plane		YCbCr plane	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Lena	38.41	0.9964	38.48	0.9933	42.19	0.9983	43.45	0.9978	39.74	0.9975	39.70	0.9949
Milkdrop	43.09	0.9972	40.51	0.9974	42.98	0.9971	43.79	0.9987	42.61	0.9971	42.88	0.9985
Peppers	39.97	0.9973	39.81	0.9948	40.95	0.9981	42.87	0.9976	38.86	0.9968	38.85	0.9943
Sailboat	39.05	0.9933	38.42	0.9943	39.71	0.9951	42.37	0.9977	37.42	0.9916	35.99	0.9901
Plane	37.19	0.9751	37.55	0.9905	43.21	0.9809	43.72	0.9977	36.64	0.9737	37.12	0.9911
Girl	40.90	0.9849	42.18	0.9963	41.85	0.9861	43.39	0.9963	41.01	0.9839	41.92	0.9955
Baboon	38.58	0.9953	39.78	0.9932	34.44	0.9877	39.00	0.9898	32.10	0.9791	33.46	0.9635
House	35.09	0.9832	36.37	0.9852	39.94	0.9922	42.61	0.9957	33.87	0.9805	34.77	0.9756
Tiffany	39.88	0.9960	40.71	0.9962	41.58	0.9969	43.36	0.9984	39.56	0.9953	40.14	0.9964

Table 7 Subjective quality of the proposed method on different planes of the restored color medical images

Images	Attacks											
	Crop				Text addition				Copy Paste			
	RGB plane		YCbCr plane		RGB plane		YCbCr plane		RGB plane		YCbCr plane	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Medical1	38.72	0.9116	39.57	0.9869	42.70	0.9234	43.49	0.9964	38.377	0.9069	39.25	0.9846
Medical2	38.92	0.9409	39.79	0.9914	41.80	0.9516	43.12	0.9966	37.45	0.9353	38.45	0.9871
Medical3	38.99	0.9172	39.66	0.9958	44.54	0.9150	43.83	0.9990	44.30	0.9271	44.12	0.9990
Medical4	44.04	0.9699	43.91	0.9896	44.03	0.9705	43.99	0.9901	43.52	0.9693	43.51	0.9887
Medical5	41.21	0.9033	41.68	0.9942	43.86	0.9101	43.73	0.9979	41.17	0.9039	41.59	0.9950
Medical6	39.26	0.9021	39.85	0.9841	44.52	0.9150	43.80	0.9957	40.96	0.9119	41.28	0.9895
Medical7	41.21	0.9033	41.68	0.9942	43.86	0.9101	43.73	0.9979	41.18	0.9039	41.59	0.9950
Medical8	39.76	0.9520	40.45	0.9926	44.37	0.9614	44.15	0.9979	43.29	0.9485	43.26	0.9973
Medical9	36.72	0.9744	37.73	0.9915	44.38	0.9821	43.38	0.9978	34.39	0.9652	35.52	0.9868
Medical10	41.42	0.9015	41.38	0.9950	44.92	0.9034	43.61	0.9983	42.14	0.9066	42.15	0.9963
Medical11	38.29	0.9735	39.58	0.9945	40.87	0.9776	43.14	0.9986	35.33	0.9665	36.53	0.9912
Medical12	41.63	0.9658	42.03	0.9940	43.85	0.9731	43.93	0.9964	41.14	0.9619	41.62	0.9924
Medical13	39.98	0.9598	40.64	0.9908	42.88	0.9653	43.64	0.9969	39.10	0.9584	39.88	0.9909
Medical14	38.43	0.9263	39.24	0.9875	41.02	0.9402	43.00	0.9967	36.40	0.9329	37.41	0.9896
Medical15	37.23	0.9317	38.10	0.9875	44.37	0.9438	43.95	0.9982	39.20	0.9370	39.94	0.9917
Medical16	34.52	0.8912	35.63	0.9811	44.52	0.9107	43.77	0.9989	35.89	0.9030	36.88	0.9885
Medical17	41.25	0.9705	41.73	0.9922	43.04	0.9794	43.69	0.9963	39.21	0.9658	39.97	0.9897
Medical18	43.15	0.9675	43.22	0.9950	44.05	0.9724	43.95	0.9958	42.95	0.9655	43.08	0.9951
Medical19	39.26	0.9730	40.03	0.9933	43.28	0.9802	43.86	0.9981	37.89	0.9684	38.80	0.9907
Medical20	42.93	0.9078	42.97	0.9990	41.59	0.9120	43.31	0.9992	36.66	0.9073	37.66	0.9959
Medical21	38.64	0.9602	39.49	0.9945	43.09	0.9703	43.71	0.9985	37.93	0.9549	38.85	0.9934
Medical22	38.75	0.9240	39.62	0.9839	42.93	0.9356	43.65	0.9955	37.96	0.9274	38.91	0.9865
Medical23	40.19	0.9755	40.84	0.9859	40.01	0.9830	42.52	0.9939	35.81	0.9701	36.82	0.9811
Medical24	42.58	0.9603	42.81	0.9929	43.71	0.9660	43.88	0.9956	41.86	0.9561	42.23	0.9908
Medical25	39.27	0.9693	40.05	0.9923	43.27	0.9801	43.76	0.9977	39.08	0.9680	39.87	0.9906
Medical26	40.61	0.9695	41.23	0.9949	43.88	0.9765	43.95	0.9979	37.61	0.9627	38.56	0.9905
Medical27	40.78	0.9664	41.44	0.9915	40.56	0.9759	42.15	0.9954	36.97	0.9665	38.04	0.9904

Table 7 (continued)

Images	Attacks											
	Crop				Text addition				Copy Paste			
	RGB plane		YCbCr plane		RGB plane		YCbCr plane		RGB plane		YCbCr plane	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Medical28	41.96	0.9463	42.26	0.9976	40.44	0.9514	42.63	0.9986	36.60	0.9412	37.56	0.9952
Medical29	37.03	0.9124	38.35	0.9881	43.41	0.9263	43.93	0.9983	38.57	0.9149	39.55	0.9912
Medical30	39.93	0.9656	40.44	0.9940	45.15	0.9768	44.25	0.9989	40.00	0.9666	40.51	0.9950
Medical31	42.70	0.9694	42.89	0.9941	43.99	0.9752	43.96	0.9958	42.78	0.9672	42.94	0.9947
Medical32	41.11	0.9694	40.82	0.9898	43.54	0.9783	43.65	0.9960	38.61	0.9653	39.60	0.9885
Medical33	43.93	0.9664	43.83	0.9962	44.02	0.9673	43.98	0.9967	43.26	0.9643	43.53	0.9960
Medical34	42.98	0.9679	43.13	0.9912	43.41	0.9723	43.82	0.9929	42.02	0.9637	42.42	0.9888
Medical35	43.37	0.9366	43.55	0.9938	43.66	0.9389	43.84	0.9952	43.22	0.9357	43.41	0.9931
Medical36	41.58	0.9682	41.99	0.9903	43.42	0.9758	43.84	0.9949	40.70	0.9636	41.25	0.9891
Medical37	43.97	0.9674	43.75	0.9927	44.17	0.9693	44.02	0.9941	43.28	0.9646	43.34	0.9924
Medical38	43.15	0.9682	43.25	0.9912	43.79	0.9725	43.93	0.9927	42.43	0.9645	42.70	0.9892
Medical39	41.10	0.9466	41.62	0.9963	42.28	0.9522	43.36	0.9975	38.86	0.9389	39.68	0.9940
Medical40	39.94	0.9285	40.61	0.9932	42.28	0.9378	43.32	0.9976	37.94	0.9260	38.84	0.9900
Medical41	43.23	0.9694	43.22	0.9895	44.36	0.9745	44.09	0.9933	42.21	0.9673	42.87	0.9886
Medical42	43.45	0.9690	43.50	0.9919	43.85	0.9718	43.94	0.9927	42.95	0.9668	43.13	0.9911
Medical43	43.94	0.9672	43.86	0.9951	44.02	0.9678	43.98	0.9954	43.78	0.9663	43.74	0.9952
Medical44	43.88	0.9664	43.79	0.9942	44.06	0.9677	43.98	0.9947	43.61	0.9647	43.58	0.9943
Medical45	40.64	0.9633	41.34	0.9906	44.99	0.9742	44.25	0.9972	41.26	0.9664	41.57	0.9936
Medical46	41.52	0.9703	42.21	0.9952	42.78	0.9770	43.63	0.9977	39.41	0.9678	40.17	0.9937
Medical47	42.37	0.9720	42.64	0.9978	43.76	0.9769	43.95	0.9986	41.43	0.9684	41.44	0.9974
Medical48	43.23	0.9684	43.30	0.9924	43.78	0.9724	43.92	0.9949	42.41	0.9654	42.66	0.9917
Medical49	36.82	0.9112	37.72	0.9924	44.37	0.9224	43.84	0.9991	38.27	0.9195	39.04	0.9951
Medical50	42.00	0.9639	42.34	0.9878	43.59	0.9703	43.84	0.9926	41.21	0.9607	41.49	0.9852

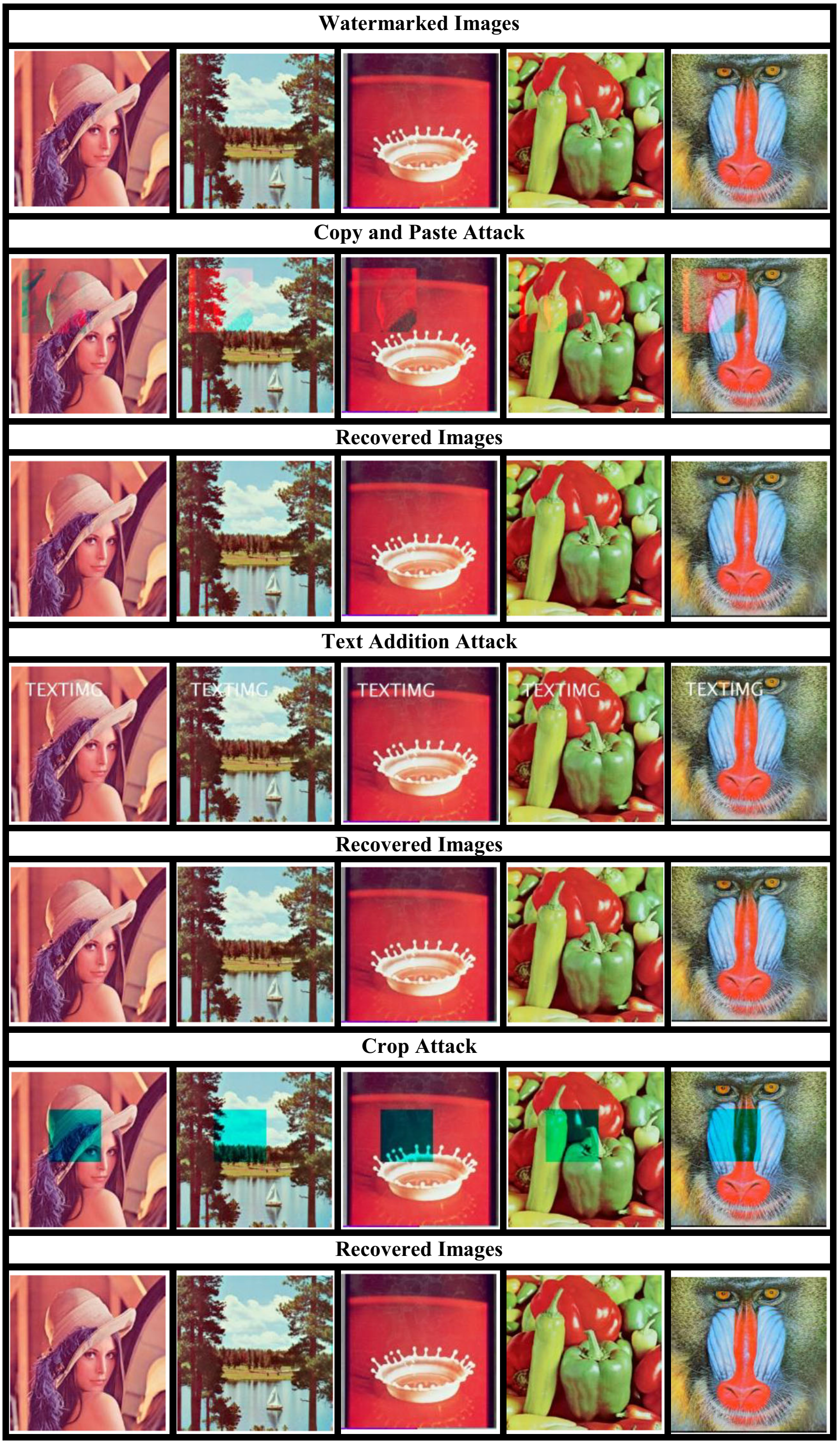


Fig. 7 Tamper detection and recovery in general color images under several image attacks

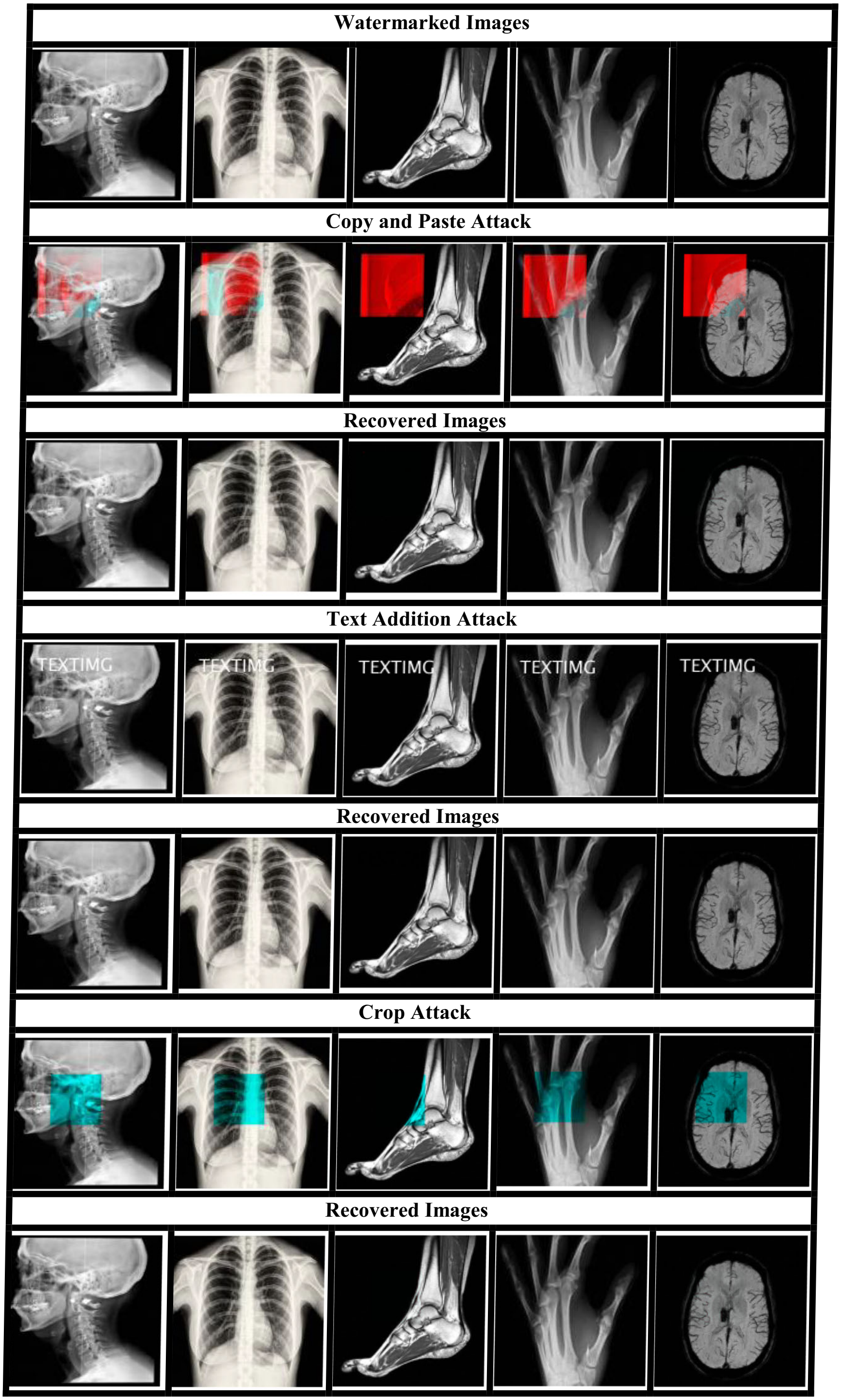
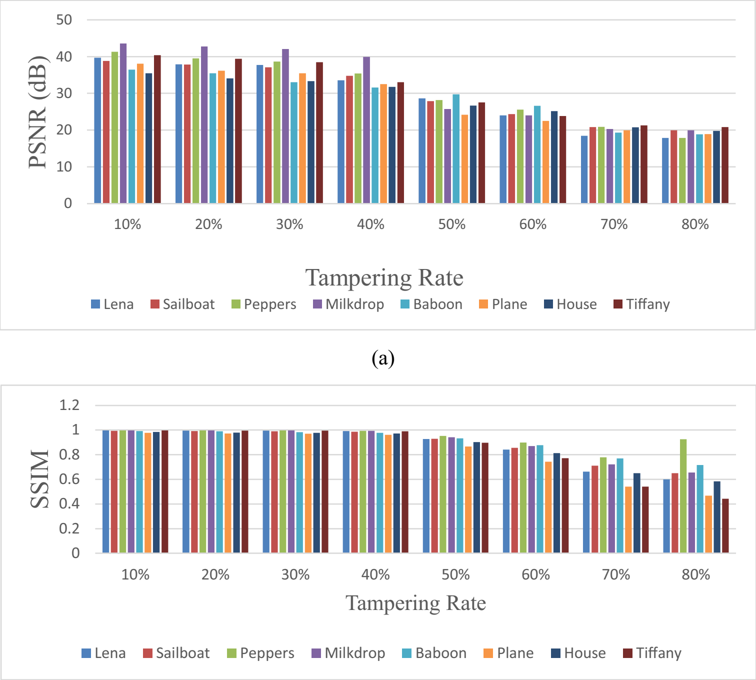


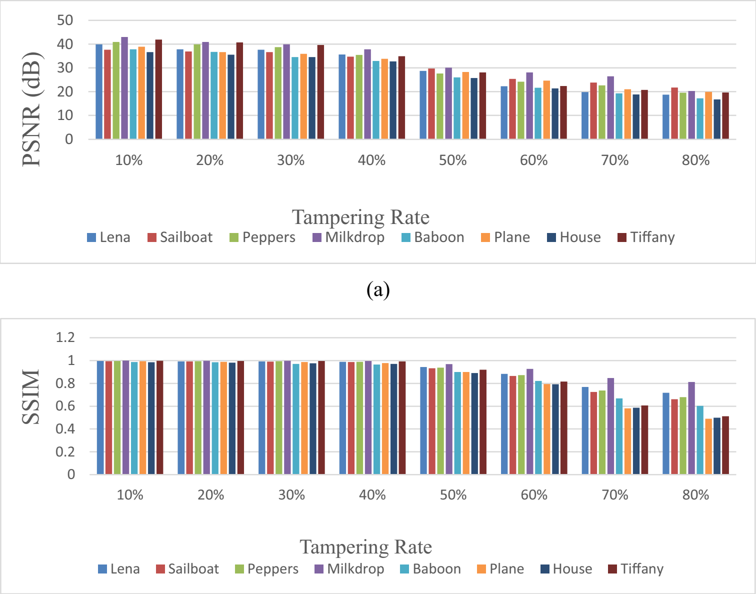
Fig. 8 Tamper detection and recovery in medical color images under several image attacks



(a)

(b)

Fig. 9 **a** PSNR performance of different tampering rates for RGB color image plane, **b** SSIM performance of different tampering rate for RGB color image plane



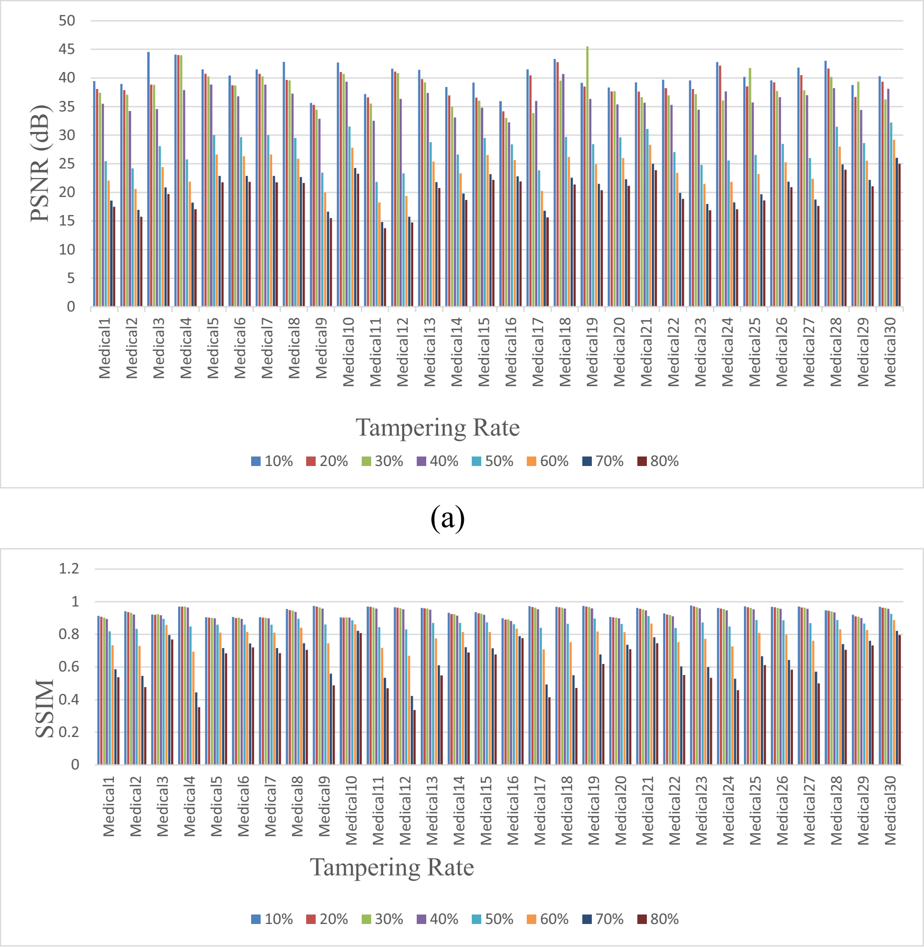
(a)

(b)

Fig. 10 **a** PSNR performance of different tampering rates for YCbCr color image plane for general images, **b** SSIM performance of different tampering rates for YCbCr color image plane for general images medical planes with different amounts of tampering rate i.e. 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%. The deviation of PSNR and SSIM with a change in the percentage of the tampered region in an image plane (RGB, YCbCr) has been presented in graphs in Figs. 9 and 10.

Figures 11 and 12 show the PSNR and SSIM of the color medical image planes when imperiled to different tampering rates. The proposed method shows the competence and usefulness after comparing it with other state of art techniques. The consequences show the dominance of the proposed method in terms of average PSNR and average SSIM. These results are shown in Tables 8 and 9 after comparing them with other existing techniques. Further, a comparison of the proposed method with a scheme reported in [6] for Kodak images taken from the Kodak database [26], which consists of 24 images of size 512 × 768 has been shown in Fig. 13a and b.

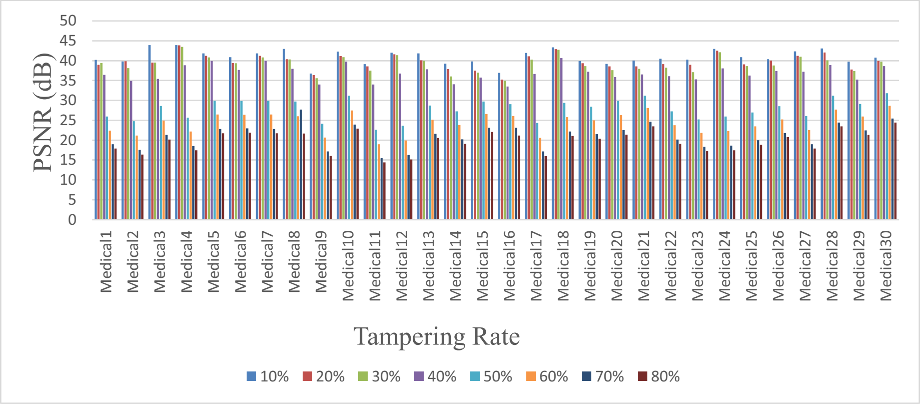
In the proposed method color images from the UCID image database [30] have been used for generalizing the experimentation of the proposed method. Figures 14, 15 shows the PSNR



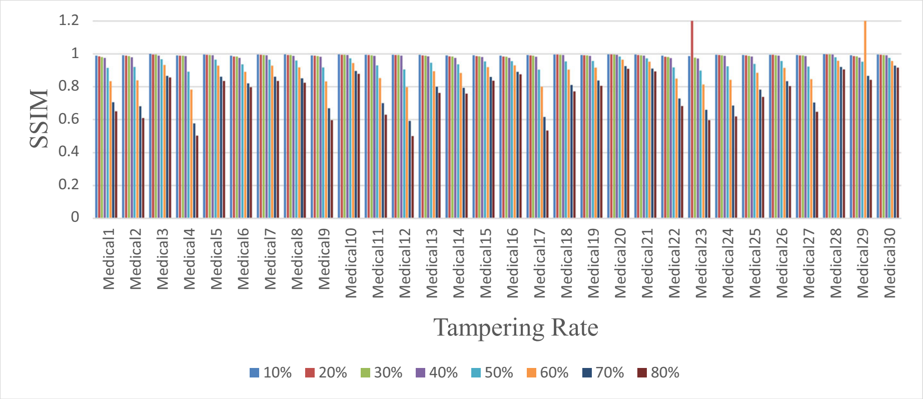
(a)

(b)

Fig. 11 **a** PSNR performance of different tampering rates for RGB color image plane for medical images, **b** SSIM performance of different tampering rates for RGB color image plane for medical images



(a)



(b)

Fig. 12 **a** PSNR performance of different tampering rates for YCbCr color image plane for medical images, **b** SSIM performance of different tampering rates for YCbCr color image plane

and SSIM values of the UCID watermarked color images in different planes (RGB, YCbCr). The results were taken after being subjected to various attacks like crop, text addition, and copy and paste attacks are presented in the tabular form in Table 10. Figures 16 and 17

Table 8 Comparison of average PSNR (dB) of the proposed method with other existing techniques

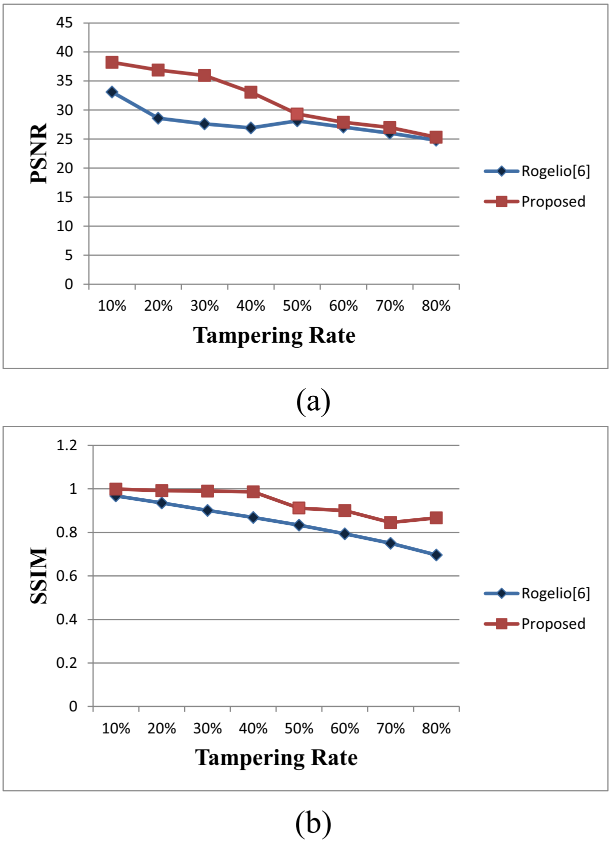
	Tampering Rate							
	10%	20%	30%	40%	50%	60%	70%	80%
Singh [49]	26.55	21.47	18.27	15.96	14.16	12.59	11.29	10.23
Dadkhan [11]	22.51	17.32	14.52	12.64	11.40	10.39	09.61	09.03
Tong [52]	35.07	28.29	23.84	20.32	17.45	15.10	13.16	11.52
Fan [12]	31.47	28.36	21.62	15.79	15.69	11.57	11.57	08.10
Tai [51]	25.89	20.57	17.43	15.21	13.54	12.01	10.80	09.81
Molina [37]	37.34	33.98	31.28	28.47	26.00	23.51	21.23	19.20
Proposed	39.22	37.89	36.97	34.07	27.32	24.50	21.35	19.25

Table 9 Comparison of average SSIM of the proposed method with other existing techniques

	Tampering Rate							
	10%	20%	30%	40%	50%	60%	70%	80%
Singh [49]	0.9290	0.8310	0.7257	0.6215	0.5139	0.3984	0.2855	0.1799
Dadkhan [11]	0.9131	0.7983	0.6855	0.5731	0.4704	0.3586	0.2506	0.1511
Tong [52]	0.9733	0.9171	0.8282	0.7150	0.5849	0.4520	0.3233	0.2042
Fan [12]	0.9731	0.9502	0.8875	0.7230	0.7202	0.4249	0.4249	0.0094
Tai [51]	0.9384	0.8443	0.7364	0.6226	0.5135	0.3899	0.2744	0.1655
Molina [37]	0.9714	0.9390	0.8977	0.8368	0.7571	0.6460	0.5157	0.3958
Proposed	0.9920	0.9899	0.9879	0.9835	0.9187	0.8339	0.6724	0.6303

represents the different UCID color image planes (RGB, YCbCr) after exposure to different multi-rate tampering attacks i.e. from 5% to 80%. Further, 600 UCID color images have been used for simplifying the proposed method.

Figure 18a and b shows the PSNR and SSIM plots of the UCID watermarked color images of the said database.



(a)

(b)

Fig. 13 **a** Comparison of average PSNR (dB) of the proposed method with other existing technique. **b** Comparison of average SSIM of the proposed method with other existing technique

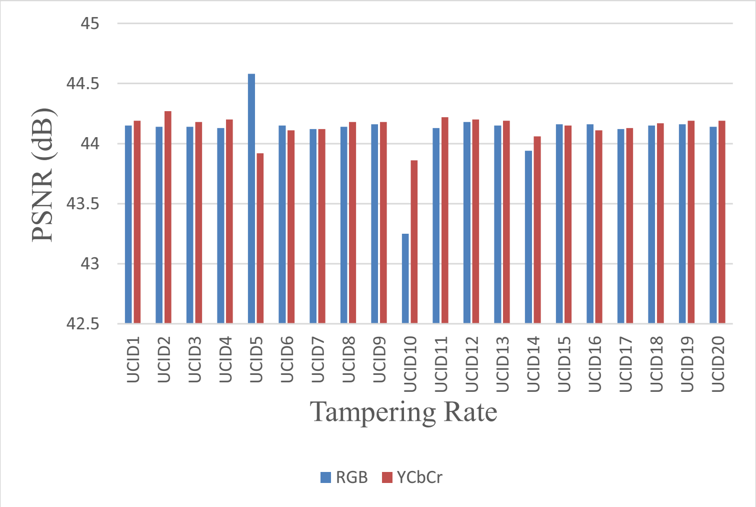


Fig. 14 PSNR of RGB and YCbCr planes

4.5 Brief discussion about the results

In this section, a brief discussion about the proposed scheme has been presented. The proposed watermarked approach is capable of tamper detection as well as recovery of the tampered color image (RGB, YCbCr) planes. In the proposed approach the color image is first separated into three planes then each plane of the color image is divided into four equal blocks. Then these four equal blocks are further sub-divided into 4 × 4 sub-blocks. The watermark is generated from these 4 × 4 corresponding sub-blocks of the image plane and is embedded in the corresponding mapping plane blocks with the location of the block where it is embedded using a chaotic algorithm. The generated 32-bit watermark is encrypted using gray code before embedding. The method is applied to all three planes of the color image. The proposed scheme

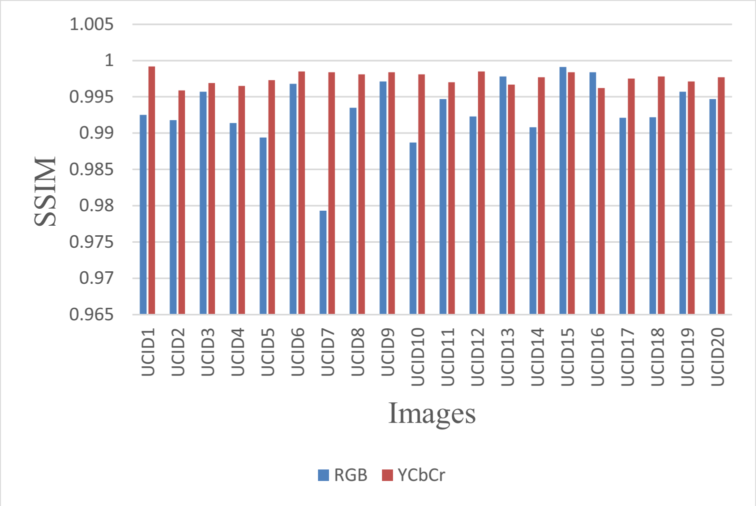
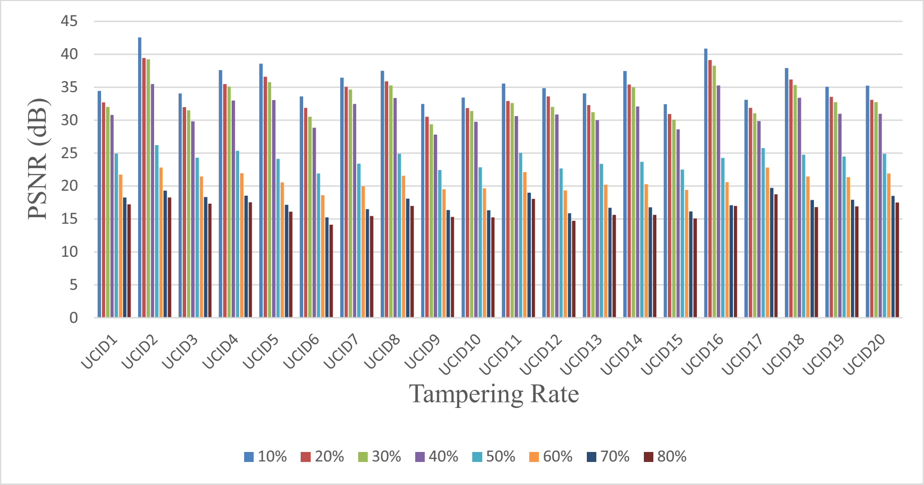


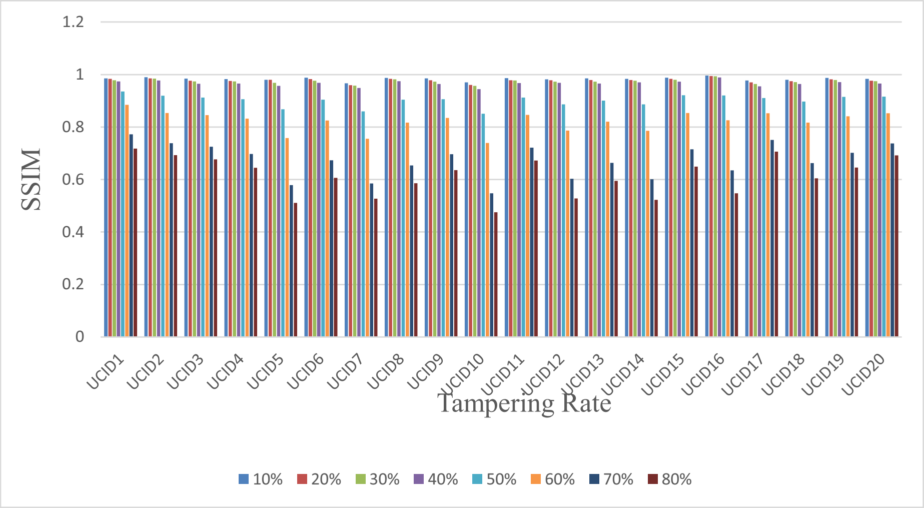
Fig. 15 SSIM of RGB and YCbCr planes

Table 10 Subjective quality of the proposed method on different planes of the color UCID images

Images	Attacks											
	Crop				Text addition				Copy Paste			
	RGB plane		YCbCr plane		RGB plane		YCbCr plane		RGB plane		YCbCr plane	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
UCID1	32.80	0.9861	33.72	0.9857	41.63	0.9906	43.36	0.9990	31.66	0.9710	34.47	0.9889
UCID2	33.66	0.9756	34.99	0.9789	44.07	0.9918	44.24	0.9959	36.12	0.9796	37.25	0.9870
UCID3	30.31	0.9701	31.80	0.9619	42.90	0.9952	43.82	0.9966	30.14	0.9700	31.61	0.9644
UCID4	33.23	0.9724	34.36	0.9761	43.62	0.9912	44.03	0.9963	34.44	0.9734	35.44	0.9749
UCID5	36.32	0.9695	38.18	0.9851	41.05	0.9862	43.13	0.9971	34.46	0.9641	36.15	0.9826
UCID6	31.30	0.9824	32.91	0.9801	37.31	0.9940	40.73	0.9970	29.39	0.9723	30.96	0.9689
UCID7	34.79	0.9546	36.45	0.9767	44.08	0.9794	44.11	0.9985	34.07	0.9563	35.86	0.9783
UCID8	35.02	0.9813	37.07	0.9902	40.46	0.9915	42.92	0.9975	33.81	0.9762	35.90	0.9887
UCID9	29.44	0.9768	31.56	0.9625	37.07	0.9939	40.95	0.9962	27.82	0.9643	30.06	0.9447
UCID10	32.36	0.9607	34.45	0.9775	40.92	0.9876	42.98	0.9973	30.91	0.9549	32.89	0.9675
UCID11	30.85	0.9676	33.08	0.9646	44.10	0.9947	44.18	0.9970	31.26	0.9731	33.50	0.9680
UCID12	35.25	0.9840	35.79	0.9882	37.16	0.9884	41.14	0.9966	31.18	0.9686	33.03	0.9734
UCID13	32.91	0.9815	34.35	0.9739	38.33	0.9944	41.76	0.9944	30.47	0.9676	32.32	0.9561
UCID14	33.74	0.9771	35.70	0.9870	42.49	0.9904	43.52	0.9976	33.76	0.9722	36.14	0.9872
UCID15	31.31	0.9862	33.10	0.9759	36.33	0.9956	40.58	0.9961	29.01	0.9754	30.65	0.9565
UCID16	37.18	0.9914	39.06	0.9843	42.97	0.9979	43.76	0.9959	36.78	0.9908	38.67	0.9836
UCID17	32.18	0.9722	33.53	0.9714	38.89	0.9883	42.08	0.9965	30.11	0.9604	31.36	0.9623
UCID18	33.82	0.9722	35.88	0.9867	40.28	0.9890	42.85	0.9968	33.98	0.9676	35.48	0.9865
UCID19	33.17	0.9812	34.53	0.9749	41.19	0.9941	43.03	0.9960	31.84	0.9764	33.45	0.9674
UCID20	30.44	0.9681	31.94	0.9662	42.09	0.9938	43.55	0.9976	31.95	0.9726	33.20	0.9671



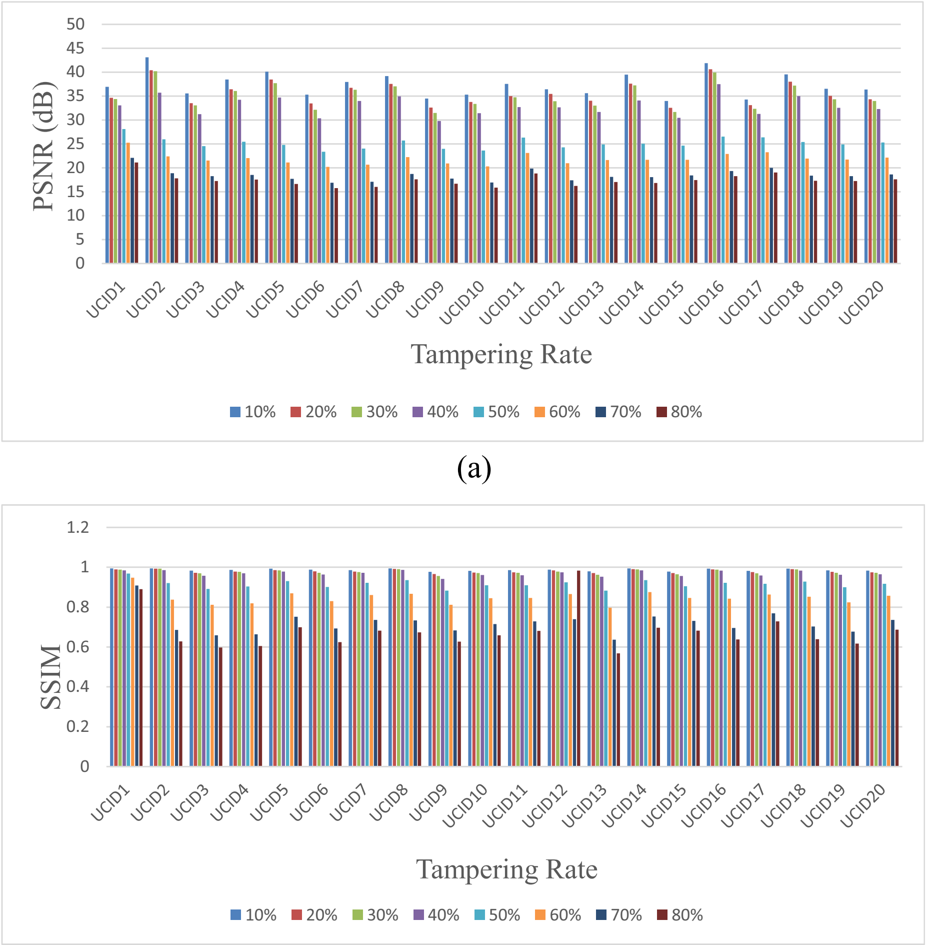
(a)



(b)

Fig. 16 **a** PSNR performance of different tampering rates for RGB color image plane for UCID images, **b** SSIM performance of different tampering rates for RGB color image plane for UCID images

shows 39.12 dB, 40.76 dB, 37.97 dB PSNR for RGB plane image, and 39.31 dB, 42.72 dB, 38.31 dB PSNR for YCbCr plane image after being subjected to various attacks like crop, text addition, and copy and paste. While imperiled to multi tampering rate attacks it shows 39.22 dB for 10%, 37.89 dB for 20% 36.97 dB for 30%, 34.07 dB for 40%, 27.32 dB for 50%, 24.50 dB for 60% etc. PSNR for color images for different tampering rates. Comparison of the proposed scheme with schemes [11–13, 37, 51, 52], and shows better results of the multi-region attacks. Also, a comparison of the proposed method with a scheme reported in [6] for Kodak images provides better results when subjected to multi tampering rate attacks. Further, the proposed scheme is implemented on the UCID



(a)

(b)

Fig. 17 **a** PSNR performance of different tampering rate for YCbCr color image plane for U0CID images, **b** SSIM performance of different tampering rate for YCbCr color image plane for UCID images

database color image planes (RGB, YCbCr) which also shows the superiority of the suggested scheme. To generalize the results, PSNR and SSIM of extra 600 UCID color images have been shown in Fig. 18a and b) Also, the presented method is found computationally efficient since it takes an average time of 16.66 s for data embedding which is decreased by 21.95 s when compared with [49].

5 Conclusion

This paper presents a watermarking framework for tamper detection and self-recovery of color image planes (RGB, YCbCr). The scheme has been evaluated for color images (general as well as medical). It has been found the proposed method displays better visual quality results as evident from the average PSNR of 39.22 dB for RGB color space and 39.37 dB for YCbCr

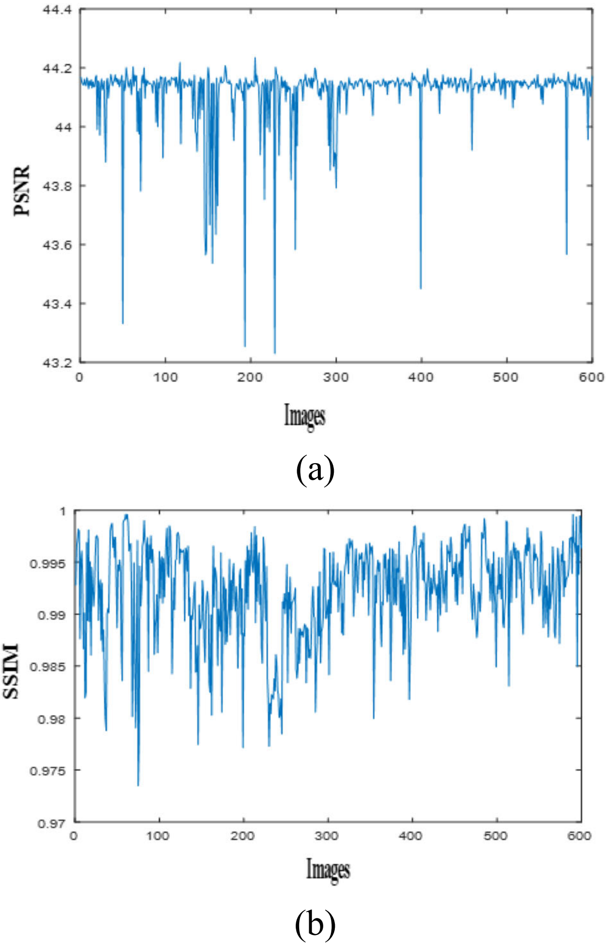


Fig. 18 **a** PSNR performance of RGB color image plane for 600 UCID images, **b** SSIM performance of RGB color image plane for 600 UCID images

color space. The self-watermarked images (with embedded recovery information) are subjected to various attacks such as cropping, copy and paste, and text addition attacks. Under all the attacked scenarios we have been successfully able to restore the damaged image blocks of the attacked images. Further, in multi-tampering rate attacks, a watermarked color image is attacked with different percentage rates i.e. 10%, 20%, 30%, 40%, 50%, 60%, 70% and 80%. Our scheme reports a significantly good range of PSNR values of 39.22 dB, 37.89 dB, and 36.97 dB for recovered images even when attacked with multi-tampering rates of 10%, 20%, and 30% respectively. This shows that our scheme is capable of recovering the tampered blocks even after multiple attacks. A comparison of our work with the state-of-art techniques shows that our method provides better restoration results with high perceptibility. The proposed method has not been tested for all image processing attacks like Gaussian noise and compression attacks. Therefore, in the future, we will try to test our scheme for these signal processing attacks as well. Further, we plan to make a standard MATLAB function file of this code and make it available via MATHWORKS.

Declarations

- The authors have no relevant financial or non-financial interests to disclose.
- The authors have no competing interests to declare that are relevant to the content of this article.
- All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript.
- The authors have no financial or proprietary interests in any material discussed in this article.

References

1. Alsmirat AM, Alem FA, Ayyoub MA, Jararweh Y, Gupta BB (2019) Impact of digital fingerprint image quality on the fingerprint recognition accuracy. Multimed Tools Appl 78:3649–3688. https://doi.org/10.1007/s11042-017-5537-5
2. Anand A, Singh AK (2020) An improved DWT-SVD domain watermarking for medical information security. Comput Commun 152:72–80. https://doi.org/10.1016/j.comcom.2020.01.038
3. Ansari IA, Pant M, Ahn CW (2015) SVD based fragile watermarking scheme for tamper localization and self-recovery. Int J Mach Learn Cybern 7(6):1225–1239
4. Azeroual A, Karim A (2017) Real-time image tamper localization based on fragile watermarking and Faber-Schauder wavelet. *AEU-Int J Electron Commun* 79:207–218
5. Bhalerao S, Ansari IA, Kumar AA (2020) Secure image watermarking for tamper detection and localization. J Ambient Intell Humaniz Comput 12:1057–1068. https://doi.org/10.1007/s12652-020-02135-3
6. Rogelio RR, Cruz-Ramos C, Ponomarvoy V, Garcia-Salgado BP et al., (2021) Color image self recovery and tampering detection scheme based on fragile watermarking with high recovery capability. *Applies Sciences,* 11(7): 3189
7. Bravo SS, Calderon F, Li TC, Nandi AK (2020) Fast fragile watermark embedding and iterative mechanism with high self-restoration performance. *Digital Signal Processing* 73:83–92
8. Cao F et al (2017) Hierarchical recovery for tampered images based on watermark self-embedding. *Displays* 46:52–60
9. Chamlawi R, Khan A, Usman I (2010) Authentication and recovery of images using multiple watermarks. *Comput Electr Eng* 36(3):578–584
10. Dadkhah S, Manaf AA, Sadeghi S (2014) An efficient image self-recovery and tamper detection using fragile watermarking. *International Conference Image Analysis and Recognition*. Springer, Cham, 504–513
11. 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
12. Fan MQ, Wang HX (2018) An enhanced fragile watermarking scheme to digital image protection and self-recovery. Signal Process Image Commun 66:19–29
13. Feng B, Li X, Jie Y, Guo C, Fu H (2020) A novel semi-fragile digital watermarking scheme for scrambled image authentication and restoration. Mobile Netw Appl 25:82–94. https://doi.org/10.1007/s11036-018-1186-9
14. Gull S, Parah SA, Khan M (2020) Reversible data hiding exploiting Huffman encoding with dual images for IoMT based healthcare. Comput Commun 163:134–149
15. Gull S, Loan NA, Parah SA, Sheikh JA, Bhat GM (2020) An efficient watermarking technique for tamper detection and localization of medical images. J Ambient Intell Humaniz Comput 11(5):1799–1808
16. Gull S, Mansour RF, Aljehane NO, Parah SA (2021) A self-embedding technique for tamper detection and localization of medical images for smart-health. Multimed Tools Appl 80(19):29939–29964
17. Hassan FS, Gutub AA (2021) Improving data hiding within colour images using hue component of HSV colour space. CAAI Trans Intell Technol *7:56–68*. https://doi.org/10.1049/cit2.12053
18. He H, Chen F, Tai HM, Kalker T, Zhang J (2011) Performance analysis of a block-neighborhood-based self-recovery fragile watermarking scheme. IEEE Trans Inf Forensics Secur 7(1):185–196
19. Huang SC, Jiang CF (2012) A color image authentication and recovery method using block truncation code embedding. J Mar Sci Technol 20(1):49–55
20. Hurrah NN, Parah SA, Loan NA, Sheikh JA, Elhoseny M, Muhammad K (2018) Dual watermarking framework for privacy protection and content authentication of multimedia. Futur Gener Comput Syst 94: 654–673. https://doi.org/10.1016/j.future02018.12.036
21. Hurrah NN, Parah SA, Sheikh JA (2020) Embedding in medical images: an efficient scheme for authentication and tamper localization. Multimed Tools Appl 79:21441–21470. https://doi.org/10.1007/s11042-020-08988-2

22. Hussan M et al (2021) Tamper detection and self-recovery of medical imagery for smart health. Arab J Sci Eng:1–17. https://doi.org/10.1007/s13369-020-05135-9AJSE-D-20-03588R1
23. Kabir MA (2021) An efficient low bit rate image watermarking and tamper detection for image authentication. SN Appl Sci 3:400. https://doi.org/10.1007/s42452-021-04387-w
24. Kiatpapan S, Kondo T (2015) An image tamper detection and recovery method based on self-embedding dual watermarking. *2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*. IEEE, 1–6
25. Kishore RR (2020) A novel and efficient blind domain image watermarking in transform domain. Procedia Comput Sci 167:1505–1514
26. Kodak image dataset is available on http://r0k.us/graphics/kodak. Accessed 7 June 2021
27. Korus P, Dziech A (2012) Efficient method for content reconstruction with self-embedding. IEEE Trans Image Process 22(3):1134–1147
28. Korus P, Dziech A (2013) Adaptive self-embedding scheme with controlled reconstruction performance. IEEE Trans Inf Forensics Secur 9(2):169–181
29. Kumar A, Ghrera SP, Tyagi V (2017) An ID-based secure and flexible buyer-seller watermarking protocol for copyright protection.
30. UCID Image Dataset is available on http://homepages.lboro.ac.uk/cogs/datasets/ucid/data/ucid.v2.tar.gz. Accessed 24 Aug 2020
31. Lee TY, Shinfeng DL (2008) Dual watermark for image tamper detection and recovery. Pattern Recogn 41(11):3497–3506
32. Li Y, Song W, Zhao X, Wang J, Zhao L (2019) A novel image tamper detection and self-recovery algorithm based on watermarking and chaotic system. Mathematics 7:955. https://doi.org/10.3390/math7100955
33. Lin ET, Christine IP, Edward JD (2000) Detection of image alterations using semifragile watermarks. Security and Watermarking of Multimedia Contents II (Vol. 3971, pp. 152-163). International Society for Optics and Photonics
34. Lin PL, Chung KH, Po WH (2005) A hierarchical digital watermarking method for image tamper detection and recovery. Pattern Recogn 38(12):2519–2529
35. Loan NA, Hurrah NN, Parah SA, Lee JW, Sheikh JA, Bhat GM (2018) Secure and robust digital image watermarking using coefficient differencing and chaotic encryption. IEEE Access 6:19876–19897
36. Mehta R, Rajpal N, Virendra PV (2018) Robust image watermarking scheme in lifting wavelet domain using GA-LSVR hybridization. Int J Mach Learn Cybern 9(1):145–161
37. Molina GJ et al (2020) An effective fragile watermarking scheme for color image tampering detection and self-recovery. Signal Process Image Commun 81:115725
38. Mustafa M, Mustafa G, Parah SA, Sheikh JA (2010) Field programmable gate array (FPGA) implementation of novel complex PN-code-generator-based data scrambler and descrambler. Maejo Int J Sci Technol 4:125–135
39. Parah SA, Bhat GM, Sheikh JA (2014) A secure and efficient data hiding technique based on pixel adjustment. *Am J Eng Technol Res* 14:38–44
40. Patra B, Patra JC (2012) Crt-based fragile self-recovery watermarking scheme for image authentication and recovery. *2012 international symposium on intelligent signal processing and communications systems*. IEEE, 430–435
41. Prasad S, Pal AK (2020) A tamper detection suitable fragile watermarking scheme based on novel payload embedding strategy. Multimed Tools Appl 79:1673–1705. https://doi.org/10.1007/s11042-019-08144-5
42. Qi X, Xin XA (2015) Singular-value-based semi-fragile watermarking scheme for image content authentication with tamper localization. J Visual Commun 30:312–327
43. Qian Z, Feng G, Zhang X, Wang S (2011) Image self-embedding with high-quality restoration capability. Digit Signal Process 21(2):278–286
44. Rajput V, Ansari IA (2019) Image tamper detection and self-recovery using multiple median watermarking. Multimed Tools Appl 79(47):35519–35535. https://doi.org/10.1007/s11042-019-07971-w
45. Saeed S, Mohammad AA (2015) A source-channel coding approach to digital image protection and self-recovery. IEEE Trans Image Process 24(7):2266–2277
46. Sahu AK (2021) A logistic map based blind and fragile watermarking for tamper detection and localization in images. J Ambient Intell Humaniz Comput. https://doi.org/10.1007/s12652-021-03365-9
47. Sarreshtedari S, Akhaee MA (2015) A Source-Channel coding approach to digital image protection and self-recovery. IEEE Trans Image Process 24:2266–2277
48. Shivendra S, Singh D, Agarwal S (2013) DCT based approach for tampered image detection and recovery using block wise fragile watermarking scheme. *Iberian Conference on Pattern Recognition and Image Analysis*. Springer, Berlin, Heidelberg, 640–647
49. Singh D, Singh SK (2017) DCT based efficient fragile watermarking scheme for image authentication and restoration. Multimed Tools Appl 76(1):953–977

50. Stergiou CL, Psannis KE, Gupta BB (1 April1, 2021) IoT-Based Big Data Secure Management in the Fog Over a 6G Wireless Network. *IEEE Internet Things J* 8(7):5164–5171. https://doi.org/10.1109/JIOT.2020.3033131
51. Tai WW, Zi JL (2018) Image self-recovery with watermark self-embedding. Signal Process Image Commun 65:11–25
52. Tong X, Liu Y, Zhang M, Chen Y (2013) A novel chaos-based fragile watermarking for image tampering detection and self-recovery. Signal Process Image Commun 28(3):301–308
53. Yu C, Li J, Li X, Ren X, Gupta BB (2018) Four-image encryption scheme based on quaternion Fresnel transform, chaos and computer generated hologram. *Multimed Tools Appl* 77:4585–4608. https://doi.org/10.1007/s11042-017-4637-6
54. Zhang X, Xiao Y, Zhao Z (2015) Self-embedding fragile watermarking based on DCT and fast fractal coding. Multimed Tools Appl 74(15):5767–5786