IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 24, 2020, accepted October 25, 2020, date of publication November 3, 2020,
date of current version November 13, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3035428

Blind Image Watermarking for Localization and Restoration of Color Images

RISHI SINHAL[1], (Graduate Student Member, IEEE), IRSHAD AHMAD ANSARI[1], (Member, IEEE), AND CHANG WOOK AHN[2], (Member, IEEE)

[1]Department of Electronics and Communication Engineering, PDPM Indian Institute of Information Technology, Design and Manufacturing Jabalpur, Jabalpur 482005, India

[2]AI Graduate School, Gwangju Institute of Science and Technology, Gwangju 61005, South Korea

Corresponding author: Chang Wook Ahn (cwan@gist.ac.kr)

This work was supported by the Institute of Information and Communications Technology Planning and Evaluation (IITP) Grant funded by the Korean Government (MSIT), Artificial Intelligence Graduate School Program (GIST), under Grant 2019-0-01842.

ABSTRACT Digital images have become easy to generate and share with tremendous growth in communication technology. Therefore, the threat of forgery and tampering in digital images has also been increased. This study proposes a blind fragile watermarking scheme for color images to provide efficient image tamper detection and self-recovery. A secret key based pseudo random binary sequence is used as a fragile watermark for tamper detection. Likewise, the recovery information is preserved in a randomized manner using a secret key. During embedding, each channel of the RGB image is divided into non-overlapping 2×4 size blocks. Each block is then watermarked using a LSB (least significant bit) replacement process in 9-base notation structure. The watermark sequence (i.e. 12-bit) for each block contains 6-bits from the fragile watermark and concatenated with the recovery information (i.e. 6 MSB (most significant bit) of block's mean value) of a different block. The experimental results confirm that the scheme is highly efficient to locate tampered region and recover the original image even in case of serious tampering. The scheme offers nearly 99% accurate tamper detection and significant recovery of tampered images (up to 80% tampering rate). Comparative results prove the significance and superiority of the scheme over existing schemes.

INDEX TERMS Blind watermarking, fragile image watermarking, tamper detection, image self-recovery.

I. INTRODUCTION

At present, it is very common to access and use digital multimedia data available over internet. Digital advancement of technological facilities makes it very easy for everyone to use digital information to fulfill their different needs [1]. A large number of digital images are transmitted over the internet. Various types of tampering/forgery are possible with these images using easily available image processing tools [2], [3]. Therefore, tamper detection and localization of digital images have become a prime concern to protect the authenticity of images [4]. Digital image watermarking is widely used to detect tampering in the images [5].

In digital image watermarking [6], [7], some information (i.e. watermark) is inserted into the digital image during embedding procedure. During extraction, the watermark is extracted from the image for different applications. Digital watermarking is used for a number of applications like copyright protection, tamper detection, self-recovery etc. Watermarking methods can be divided into three types named as blind, non-blind and semi-blind watermarking based on extraction type [8]. There is no need of host and watermark at the time of extraction in blind watermarking except the secret key [9]. In the semi-blind scheme, the information of the watermark signal and the secret key is needed during extraction. The non-blind watermarking technique needs both host and watermark along with the secret key for the extraction process. In another way, watermarking can be classified as robust, fragile or semi-fragile based on the nature of watermark information [10]. Robust watermarking is preferred for ownership or data protection applications. The semi-fragile method has features of robust as well as fragile watermarking. It survives against robust and fragile attacks like noising [11], filtering [12], compression [13],

The associate editor coordinating the review of this manuscript and approving it for publication was Liangtian Wan.

tampering, etc. to a certain limit. The fragile watermark is used to detect tampering/modification and also used to authenticate the host image. In fragile watermarking, least significant bits are modified based on a specified algorithm that results in significant visual quality as compared to the robust watermarking [14]–[16]. There are many digital image watermarking methods proposed in the past to identify tampered/modified pixels, authenticate the image, and to recover the host image. The literature review is provided in the next section. The contribution of the proposed work is as follows:

- Excellent self-recovery for even highly tampered (∼80%) images.
- Accurate tamper detection & localization (∼99%) without any compromise in self-recovery ability.
- Efficient detection & recovery against block based attacks.
- Superior performance as compared to various existing methods.

The rest of the work is drafted as follows. Section II discusses the existing fragile watermarking methods. Section III explains the proposed watermarking method. Section IV presents experimental results and discussion. At last, the work is concluded in section V.

II. RELATED WORK

Previously, Zhang and Wang [5] proposed a watermarking technique for tamper localization and restoration of gray images. The scheme used DE (difference expansion) approach to embed fragile watermark and reference bits (recovery data) into the cover image. The scheme reported tamper localization and recovery results for very low tampering rate (<3.2%). It signified that the scheme failed to perform effectively for tampering rate greater than 3.2%. Therefore, the scheme has limited applications and lacks practicability in case of severe tampering/forgery. Zhang *et al.* [17] offered two watermarking schemes based on the reference sharing process for tamper detection and image recovery. The data stored in the non-tampered part of the watermarked image has been used to recover the tampered portion of the image. The first scheme provided less accuracy regarding restoration capability. The second scheme divided the image into 3 levels and each level has different capability of restoration. The second scheme provided better results as compared to the first scheme. However, the results degraded at high tampering rates. He *et al.* [18] offered a fragile scheme for tamper detection and self-recovery of the image. The scheme used optimized block-neighborhood approach for tamper detection. The recovery of the tampered regions performed using reserved feature available in the non-tampered region and block averaging process. Although the results are satisfactory against different attacks, the scheme reported significant recovery for tampering rate up to 60%. For higher tampering rate, the scheme did not provide significant solutions to recover the tampered portions. Singh and Singh [19] proposed a self-embedding based image watermarking scheme using DCT (discrete cosine transform) to detect and localize the tampered part of the image. It also provided restoration of the tampered image. The block-wise division and two level detection mechanisms were used for the better localization of the tampered part. Although the scheme provided acceptable results, it was not able to recover the tampered image significantly for more than 50% tampering rate. Fan and Wang [20] presented an improved method of watermarking images to resolve the issue of protecting channel code parity bits and increase the restoration capability. The results were acceptable for low tampered area but the tamper detection and image restoration were poor at high tampering rate. Tai *et al.* [21] proposed a watermarking technique to authenticate the image and further recover the modified regions of the image. A chaotic map mechanism was used to select the blocks for storing the authentication data (i.e. fragile watermark) and the recovery information of the other block. During recovery process, the discrete wavelet transform was used in place of block averaging process. Qin *et al.* [22] presented a self-embedding image watermarking scheme using watermark information insertion non-uniformly. To get better visual results of recovered images, an improved block truncation coding (BTC) mechanism was offered named as optimal iterative BTC. The simulation results provided significant performance but the condition of restoration of the image was limited to 50% tampering rate. Liu *et al.* [23] proposed an image watermarking method for copyright protection and authentication. The fragile watermarking part of the method was based on LSB replacement in 3^n base (i.e. n=2) notational systems in RGB color space. During fragile watermarking, the set of pixels get self-authenticated. The scheme was able to tolerate different tampering attacks but the efficient performance against block-based attacks was not provided. In addition, the scheme did not provide restoration of the tampered images and only authenticate the image. Hurrah *et al.* [24] proposed a framework of color image watermarking to authenticate data and protect copyrights. The spatial domain-based fragile embedding has been done in such a way that the blocks self-authenticate themselves during the extraction process. Consequently, the authentication performance degrades against block-based attacks. Moreover, the scheme is unable to provide recovery of the tampered part of the images. Pal *et al.* [25] offered a watermarking scheme using block-wise division and local binary pattern (LBP). This scheme used two copies (i.e. *CD1* and *CD2*) of the host for watermarking. Authentication codes and watermark bits were embedded in the blocks of *CD1* and *CD2* by using the secret key. During the extraction, blocks were supposed to self-authenticate by recovering and comparing the authentication code. Therefore, the block-based attacks can reduce the authentication efficiency of the scheme to a great extent. Further, the scheme is not able to self–recover the tampered image. Molina-Garcia *et al.* [26] recently proposed a color image watermarking technique that confirms an effective tamper detection and image restoration. The host image was first partitioned into blocks; further the

authentication and recovery information for each block are calculated. Afterwards, a permutation mechanism was used to store the information concerning one block into another block. Experimental results showed the effective performance results of the scheme. However, the recovery results can be improved further using a better insertion mechanism.

As per the study of image watermarking literature, some of the fragile watermarking schemes perform poorly in terms of image recovery at higher tampering rates. Additionally, some schemes give poor image authentication results against block-wise attacks. Therefore, there is a scope for improvement in existing fragile watermarking methods. This article offers an improved watermarking scheme which provides image restoration capability along with efficient tamper detection and localization. The present work offers efficient tamper recovery of image even in case of severe (∼80%) tampering. Moreover, it also possesses the ability to deal with block based attacks without affecting the tamper detection and self-recovery ability.

III. PROPOSED FRAGILE WATERMARKING SCHEME

The proposed fragile watermarking scheme can be divided into three main parts, namely; the watermark embedding, watermark extraction and the recovery of the image. The proposed watermarking process is described in detail as follows.

A. WATERMARK EMBEDDING

During embedding, the host RGB image gets divided into non-overlapping blocks (uniform block size = 2 × 4). Next, the pseudo random number generation algorithm [27] is used to obtain the uniformly distributed random number (ranges between 0 and 1) sequence using a seed value (secret key). As per (1), each number (*num*) of this sequence is converted to either 0 or 1 (i.e. *bit_val*). Thus, the controlled randomized binary fragile watermark sequence (*W_temp*) is obtained. As 6 bit sequence would be used as a fragile watermark for each block, the length of the sequence *W_temp* is equal to 589824 bits (6 x no. of blocks).

$$bit\_val = \begin{cases} 1 & num > 0.5 \\ 0 & num \leq 0.5 \end{cases} \tag{1}$$

The 3^n-base (n=2) notational system based LSB replacement [23] would be performed in RGB color space during embedding. The fragile watermark embedding process contains following steps.

1. Generate a secret key (K_1) based random binary watermark *W_temp*. Divide *W_temp* into three equal size bit sequences namely *temp*_1, *temp*_2 and *temp*_3.
2. Apply block-wise division on the first channel of the image.
3. Calculate the average pixel value of each block and convert it into binary form (i.e. *Mean_bin*). Next, select 6 MSB bits from *Mean_bin* concerning each block.
4. Obtain the watermark sequence *W_recov* by cascading these 6 MSB bits of each block in a secret key (i.e.K_2) based secured random fashion.
5. Select *temp*_1 as the fragile watermark for the first channel.
6. Select 6 bits sequentially from both watermark sequences (i.e. *temp*_1 and *W_recov*) and cascade it to get the 12-bit sequences. Next, concatenate all the generated 12-bit sequences to get the watermark sequence *W_seq*. The length of the *W_seq* is two times of *W_recov*.
7. Select the first 2 × 4 size block where each column of the block represents a pixel unit (i.e. *U*). Therefore, the block has four units *U1*, *U2*, *U3*, and *U4*. Each unit has two pixels.
8. Sequentially select 12 bits from *W_seq* and convert it into the 9-base notation *W*_9 (s1s2s3s4).
9. Each pixel unit *U* has two pixels {*U=(p1, p2)*}. The digits s1, s2, s3, and s4 are to be embedded into the *U1*, *U2*, *U3*, and *U4* respectively. The following steps would be used to perform LSB replacement in an n-pixel unit *U* by embedding a digit's' of *W*_9.
   - Extract digit *E* using (2).

$$E = \sum_{i=1}^{n} 3^{i-1} p_i \bmod 3^n . where \text{ n=2} \tag{2}$$

   - Generate a value '*t*' to adjust 2-pixel unit U using (3), so that E=s as given below.

$$t = \left(s - E + \left\lfloor \frac{3^n - 1}{2} \right\rfloor \right) \bmod 3^n \tag{3}$$

   - *t* is changed to t' by converting *t* using 3 base notation and $t' = h_1 h_2 \ldots\ldots h_n$, where h_i is a digit of t' for $1 \leq i \leq n$.
   - Each digit value in t' is decreased by 1 to get $t'' = g_1 g_2 \ldots\ldots g_n$, where $g_i = h_i - 1$.
   - To get watermarked pixel unit U'=*(p_new1, p_new2)*, each digit of t'' is added to pixels of unit U using (4).

$$p\_new_i = p_i + g_j \; for \; 1 \leq i \leq n \& j = n - i + 1. \tag{4}$$

   - This process gets repeated for all pixel unit U of a block to get the watermarked block.
10. Repeat steps 7, 8 and 9 to perform the watermarking process on each block of the channel.
11. Similarly, Perform embedding operation on the second and third channel by using *temp*_2 and *temp*_3 as the fragile watermarks respectively.

The general block diagram of the proposed embedding procedure is shown in Fig.1.

B. WATERMARK EXTRACTION

The following steps are used to extract watermark and detect tampering in the watermarked image during the extraction process.

1. Initially, the block-wise division is done on the first channel of the watermarked image to divide it into 2 × 4 size blocks.

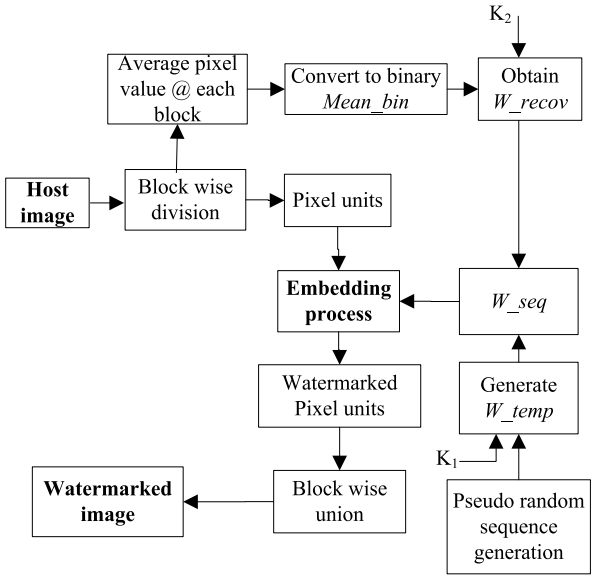


FIGURE 1. Watermark embedding process.

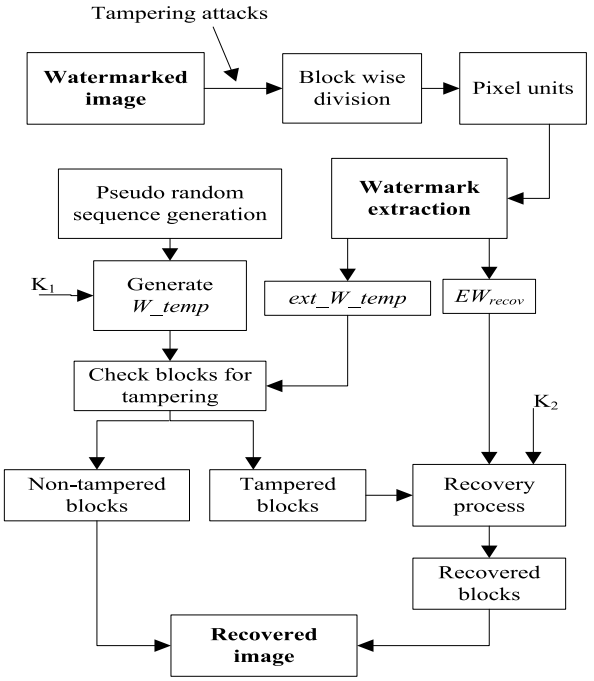


FIGURE 2. Watermark extraction process.

2. Equation (2) is applied on each pixel unit U of the first 2×4 size non-overlapping block to extract the digit ext_E. Thus the pixel-units $U1$, $U2$, $U3$, and $U4$ of a block gives ext_E1, ext_E2, ext_E3 and ext_E4 respectively.
3. Concatenate extracted digits (e.g. ext_E1 etc.) to get ext_W_9 and convert it into 12 bit binary sequence ext_W.
4. This process is repeated for each 2×4 size block sequentially.
5. Cascade the first 6-bits of ext_W of each block to get ext_W_temp. Similarly cascade the last 6 bits of ext_W of each block to get EW_{recov}.
6. Generate the watermark binary sequence W_temp using the secret key (K_1). Divide it into three equal size bit sequences namely $temp\_1$, $temp\_2$ and $temp\_3$.
7. Compare $temp\_1$ and ext_W_temp, if the analogous bits are different then the corresponding block would be considered as tampered/forged.
8. Finally, each block is marked as original or forged based on majority of non-tampered or tampered neighbor blocks respectively.

Similarly, the extraction/authentication process is repeated for the second and third channels. $temp\_2$ and $temp\_3$ are used for authentication of the second and third channels respectively. Fig.2 shows the graphical presentation of the proposed watermark extraction and recovery procedure.

Algorithm 1 Self-Recovery Process

Input: Authenticated image I_{au}, Extracted recovery information EW_{recov}, Secret key K_1.
Output: Recovered image I_{final_recov}
Assumption: $T \rightarrow$ *tampered block,* $N \rightarrow$ *the total no. of tampered blocks and* $R \rightarrow$ *the block preserving the recovery information of the tampered block* T

1. **for** m=1:3
2. Divide the m^{th} channel of I_{au} into 2×4 size blocks.
3. **for** $T = 1: N$
4. Find R
5. **if** $R \rightarrow$ not tampered
6. Select 6-bits from EW_{recov} concerning R (via K_1).
7. Add 2 LSB bits (i.e. 00) to get 8-bit bin_seq.
8. $X_{decimal}$=$bi2de\ (bin\_seq)$.
9. Replace all pixels of T with $X_{decimal}$
10. **end if**.
11. **end for**
12. **end for**
13. Obtain recovered image I_{recov}.
14. Apply smoothing process two times on the image I_{recov} to get the final recovered imageI_{final_recov}.

C. SELF-RECOVERY OF THE IMAGE

Algorithm 1 presents the self-recovery process, which has been used to recover the tampered area (i.e. blocks) of the image after the watermark extraction.

During recovery, smoothing procedure is applied two times for improved results. In the smoothing operation, the average of the pixels of neighbor blocks (non-tampered/recovered) is obtained. Afterwards, this average value is put in place of the pixel values of the tampered block. A sample result of smoothing process is shown in Fig. 3, in which a tampered block T (in yellow) is restored using smoothing operation.

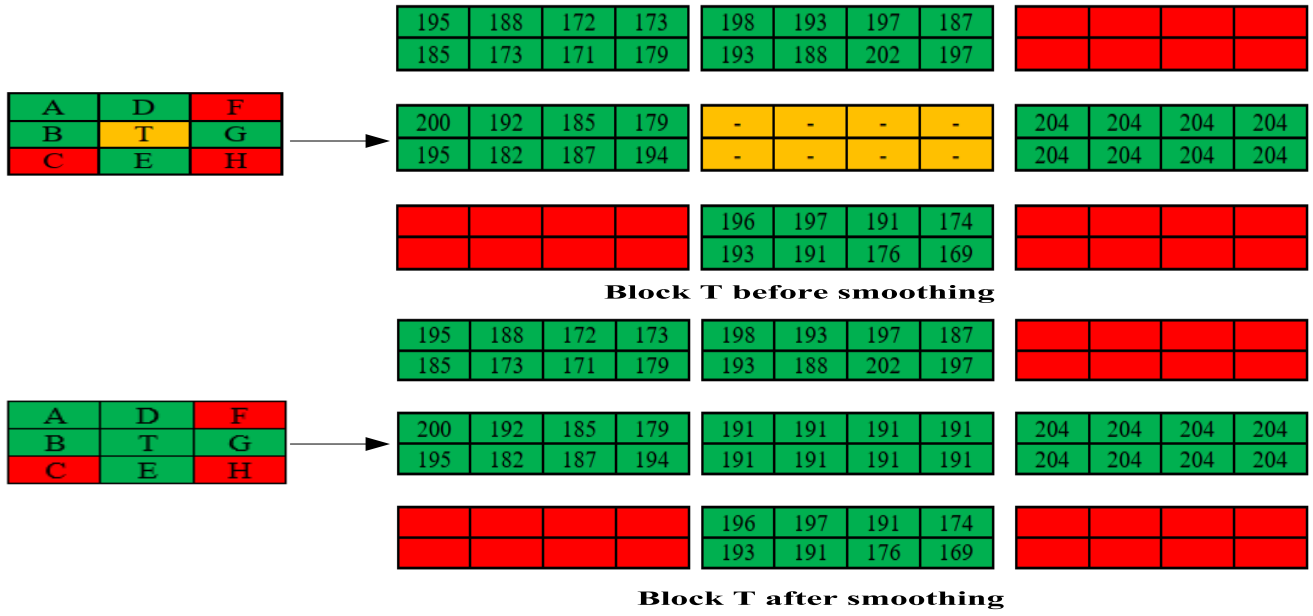


FIGURE 3. Smoothing process to recover block 'T' with the help of original and/or recovered blocks.

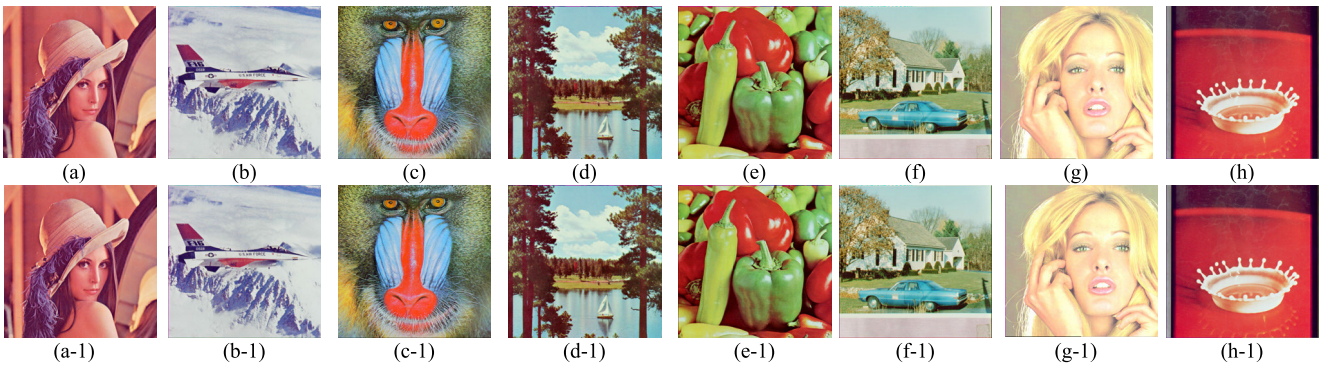


FIGURE 4. Host images (a) "Lena" (b) "Airplane" (c) "Mandrill" d) "Sailboat" (e) "Pepper" (f) "House" (g) "Tiffany" (h) "Splash." Watermarked images (a-1) "W-1" (b-1) "W-2" (c-1) "W-3" (d-1) "W-4" (e-1) "W-5" (f-1) "W-6" (g-1) "W-7" (h-1) "W-8."

The green blocks (i.e. A, B, D, E and G) are either original/recovered, whereas other tampered blocks (i.e. C, F and H) are in red color. The average of all pixels of blocks A, B, D, E and G is calculated and found to be 191 (after round off). This value is substituted into all pixel coordinates of block T. This process will be repeated for each tampered block.

IV. RESULTS AND DISCUSSION

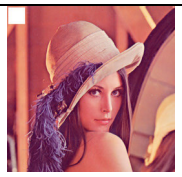
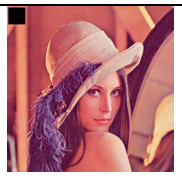
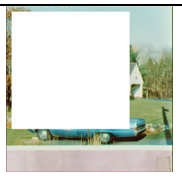
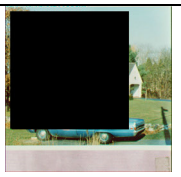
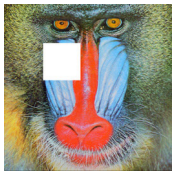
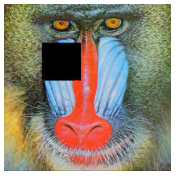
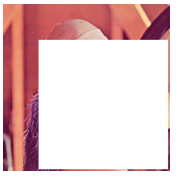
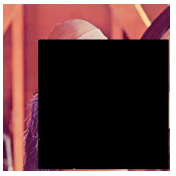
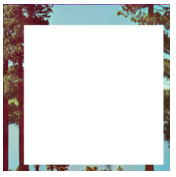
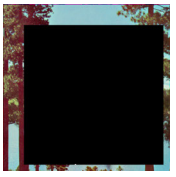
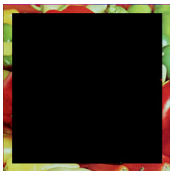
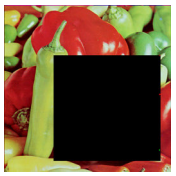
150 RGB images are used to obtain the experimental results. Though, eight standard RGB test images (size $= 512 \times 512$) have been selected from the USC-SIPI database to present the results in this article in order to compare the performance with existing schemes. These images are shown in Fig. 4. Imperceptibility parameters such as PSNR (peak signal to noise ratio) and SSIM (structural similarity index) have been evaluated to compare the visual quality between hosts and watermarked images [28]. The average parametric values in terms of PSNR and SSIM are 49.62 and 0.9986 respectively. The comparison of imperceptibility results (i.e. PSNR and SSIM) with some recently proposed state of the art methods are presented in Table 1. The results presented in Table 1 show that the variance in the imperceptibility results for different images are very less in the proposed work. This proves that the proposed method can be used for different types of images. Additionally, the results are superior to the results of other existing schemes.

The proposed scheme has been tested for different tampering attacks. The results for different tampering rates have also been obtained. The altered portion of the image has also been recovered. The obtained results show that the method can detect tampered portion very efficiently and recover the images significantly. In the proposed work, the minimum tampered area that can be detected is equal to a 2×4 size block (i.e. 8 pixels). Consequently, even if a few pixels in the block are altered, the block would be detected as the tampered block. The block size (i.e. 2×4) selection is based

TABLE 1. Comparison of proposed work with existing methods in terms of imperceptibility (host image, watermarked image).

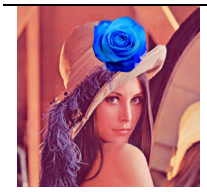
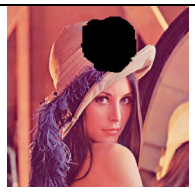
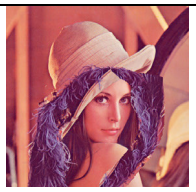
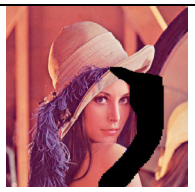
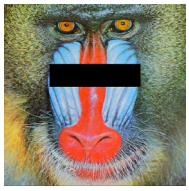
Host images	PSNR					SSIM				
	Singh [19]	Fan [20]	Tai [21]	Molina-Garcia [26]	**Proposed method**	Singh [19]	Fan [20]	Tai [21]	Molina-Garcia [26]	**Proposed method**
Lena	37.90	44.13	44.12	44.60	**49.88**	0.9307	0.9820	0.9820	0.9840	**0.9997**
Airplane	37.88	44.11	44.12	44.69	**49.88**	0.9194	0.9781	0.9781	0.9812	**0.9953**
Mandrill	37.90	44.12	44.14	44.64	**49.88**	0.9763	0.9941	0.9941	0.9947	**0.9996**
Sailboat	37.90	44.10	44.11	44.61	**49.87**	0.9493	0.9867	0.9868	0.9884	**0.9991**
Pepper	37.79	44.06	44.06	44.54	**49.70**	0.9234	0.9791	0.9791	0.9816	**0.9996**
House	37.88	44.18	44.18	44.66	**49.87**	0.9319	0.9815	0.9815	0.9834	**0.9986**
Tiffany	37.44	43.84	43.85	44.87	**48.59**	0.9246	0.9804	0.9805	0.9846	**0.9993**
Splash	37.84	44.08	44.09	44.47	**49.77**	0.8942	0.9695	0.9696	0.9737	**0.9993**

TABLE 2. Image authentication and tamper detection results for different tampering rates.

Tamper Rate (%)	Attacked image	Authenticated image	TDR_{ACC} (%)	Tamper Rate (%)	Attacked image	Authenticated image	TDR_{ACC} (%)
1	 $N_{attacked}$ =1014	 $N_{detected}$ =1013	99.901	50	 $N_{attacked}$ =49686	 $N_{detected}$ =49685	99.998
5	 $N_{attacked}$ =5220	 $N_{detected}$ =5210	99.808	60	 $N_{attacked}$ =59700	 $N_{detected}$ =59693	99.988
15	 $N_{attacked}$ =14850	 $N_{detected}$ =14847	99.979	70	 $N_{attacked}$ =68694	 $N_{detected}$ =68691	99.995
25	 $N_{attacked}$ =25155	 $N_{detected}$ =25148	99.972	80	 $N_{attacked}$ =79350	 $N_{detected}$ =79347	99.996
40	 $N_{attacked}$ =39852	 $N_{detected}$ =39847	99.987	95	 $N_{attacked}$ =93750	 $N_{detected}$ =93748	99.997

on the manual experimentation done to get an optimum size that can preserve the authentication and recovery information significantly. Before that, different block sizes such as 2×2, 2×4, 4×4 and 8×8 have been examined to get better performance. As per the embedding strategy, the embedding capacity is 1.5 bits per pixel (BPP). Thus, the block size

TABLE 3. Image authentication and tamper detection results for different types of manipulative tampering attacks.

Attacked image	Authenticated image	Tamper detection results	Attacked image	Authenticated image	Tamper detection results
		$N_{attacked}$ = 7368 $N_{detected}$ = 7355 TDR_{ACC} =99.823%			$N_{attacked}$ = 10933 $N_{detected}$ = 10908 TDR_{ACC} = 99.771%
Adding flower			**Random Copy-move**		
		$N_{attacked}$ = 6144 $N_{detected}$ = 6144 TDR_{ACC} =100%			$N_{attacked}$ = 31680 $N_{detected}$ = 31680 TDR_{ACC} =100%
Block wise copy-move			**Block wise copy-move**		
		$N_{attacked}$ = 12240 $N_{detected}$ = 12240 TDR_{ACC} =100%			$N_{attacked}$ = 18912 $N_{detected}$ = 18893 TDR_{ACC} = 99.899%
Block wise copy-paste			**Removing content**		
		$N_{attacked}$ = 13218 $N_{detected}$ = 13204 TDR_{ACC} =99.898%			$N_{attacked}$ = 2710 $N_{detected}$ = 2692 TDR_{ACC} = 99.335%
Replacing object (i.e. car)			**Multiple copy**		

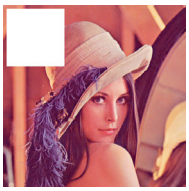
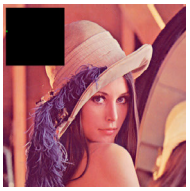
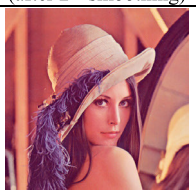
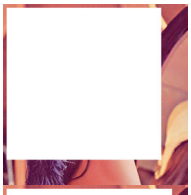
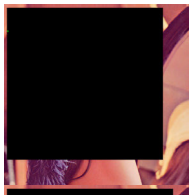
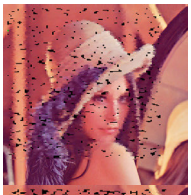
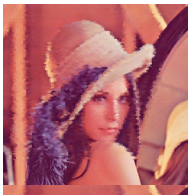
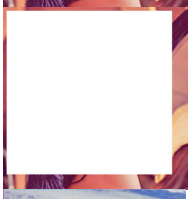
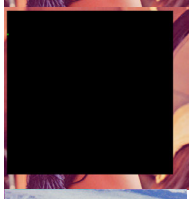
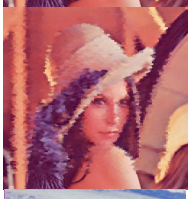
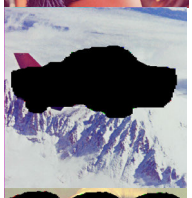
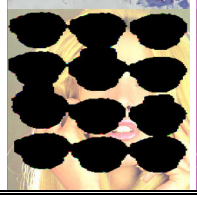
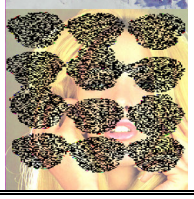
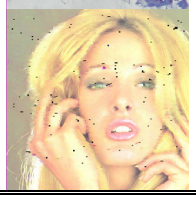
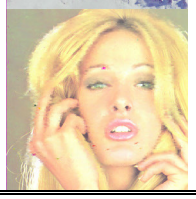
2 × 2 can store only 6 watermark bits. So, the 2 × 2 size lacks in preserving significant amount of information that would be needed for tamper detection and self-recovery during extraction process. However, the block sizes 4 × 4 and 8 × 8 provide sufficient space for storing the watermark as well as the recovery information, but the localization accuracy get reduced. It is because a complete block would be marked as tampered even if one pixel of the block is actually tampered. Since the authentication and restoration procedures are block based, a bigger block size can decrease the accurate authentication ability. It further results in poor performance in terms of self-recovery. From the imperceptibility point of view, different block sizes provide almost similar results because the embedding capacity in terms of BPP is same and equal to 1.5. However, small variations (∼0.24 dB) in PSNR values are obtained during experimentation.

The maximum tampered area that can be detected by the proposed method cover all the possible tampered blocks. As a result, tamper detection accuracy is very high. To examine the accuracy of tamper detection results of the proposed method, the tamper detection rate (TDR_{ACC}) is calculated for different tampering rates and other manipulative tampering attacks. The tamper detection rate (TDR_{ACC}) can be obtained using (5).

$$TDR_{ACC} = \frac{N_{detected}}{N_{attacked}} \times 100\ \% \tag{5}$$

where $N_{detected}$ and $N_{attacked}$ represent the number of detected tampered blocks and attacked blocks respectively. Different types of tampering attacks are employed to obtain the image experimental results. The image authentication and tamper detection results for different images have been presented in Table 2. Moreover, the results for common manipulative tampering are presented in Table 3. For better analysis, the modification and detection in each channel of an RGB color image are considered. Thus, $N_{attacked}$ presents the number of attacked blocks of all channels and $N_{detected}$ presents the number of detected tampered blocks of all channels. It is confirmed from the obtained results that the tamper detection accuracy is very high (nearly 99%). The performance of the method does not depend on the size of tampering and able to detect tampered regions efficiently.

TABLE 4. Analysis of Recovery process for different tampering rates.

Tamper rate (%)	Attacked watermarked image	Authenticated image	Recovery-1 (after extraction)	Recovery-2 (after 1st smoothing)	Final Recovery (after 2nd smoothing)	PSNR, SSIM
10						40.94, 0.9991
70						24.75, 0.9447
80						23.20, 0.9296
24.5						30.66, 0.9641
58.9						29.20, 0.9681

As shown in Table 3, the scheme is able to authenticate the image against different types of tampering including block-wise attacks such as copy-move [29] and copy-paste [30]. The scheme can effectively detect block-wise attacks because it employs different binary sequences (as a fragile watermark) for different blocks.

The recovery procedure for restoration of tampered regions of the image is depicted in the form of experimental results as shown in Table 4. Different attacks are applied on the watermarked version of the test images and the recovery image of each stage is presented, which gives an insight of the complete recovery process. It also shows the significance of smoothing procedure in order to restore the tampered portion. Smoothing process is applied two times on the authenticated image. Detected tampered region is shown as black for better understanding of recovery process for the viewers. Furthermore, the comparative study of successful recovery condition is presented in Table 5 that confirms the supremacy of the proposed scheme. The schemes [17] and [22]-B used 8×8 size blocks for watermarking. So even if a few pixels are tampered, the complete block (i.e. 64 pixels) will be considered as tampered. Likewise the schemes [22]-A and [26] used 4×4 size blocks, so if a pixel in a block gets tampered then all 16 pixels of the block are marked as tampered. Therefore, it reduces the localization accuracy. The poor localization further results in poor recovery. The scheme in [19] employed 2×2 size blocks which significantly improves the accuracy of localization and reduces the blocking artifacts.

TABLE 5. Comparative analysis with existing schemes in terms of successful recovery conditions.

Methods	PSNR (in dB)		Condition for successful recovery
	Watermarked	Recovered	
Zhang et al. [17]-A	37.9	$+\infty$	TR < 24%
Zhang et al.[17]-B	37.9	[22, 40]	TR < 66%
Singh and Singh [19]	37.8	~31	TR ≤ 50%
Qin et al. [22]-A	44.2	[33, 42]	TR < 45%
Qin et al. [22]-B	44.2	[31, 40]	TR <50%
Molina-Garcia et al. [26]	44.63	~19.20	TR ≤ 80%
Proposed scheme	49.68	~22.47	TR ≤ 80%

TABLE 6. Recovery results for different tampering rates.

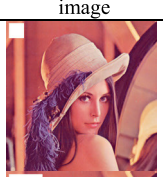
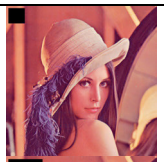
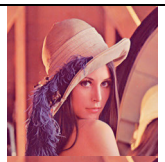
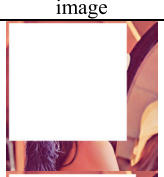
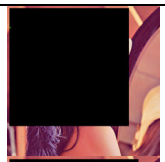
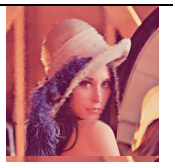
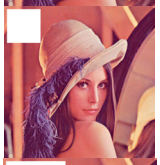
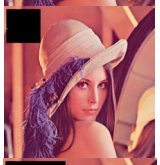
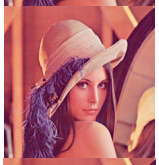
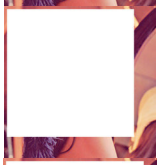
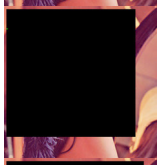
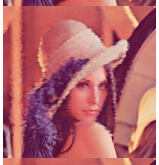
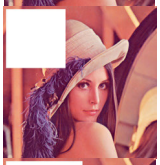
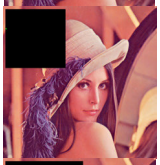
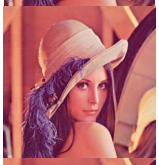
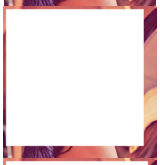
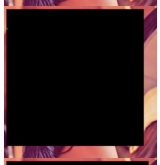
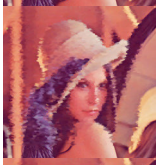
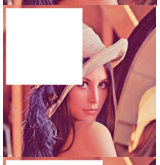
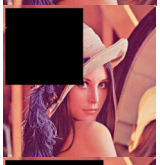
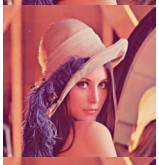
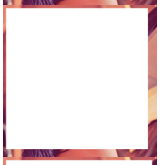
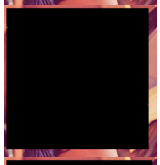
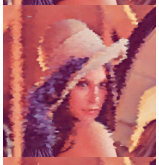
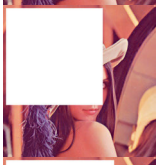
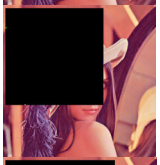
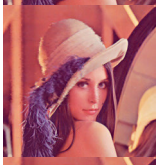
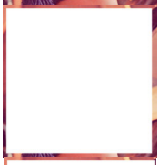
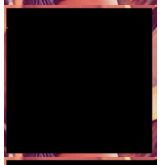
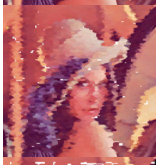
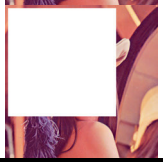
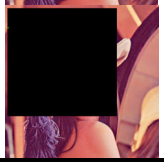
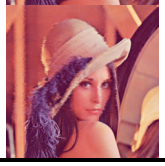
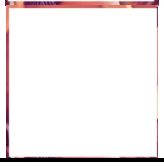
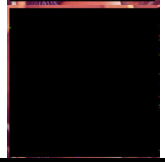
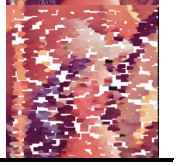
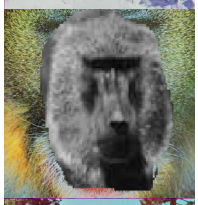
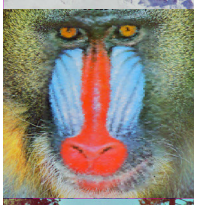
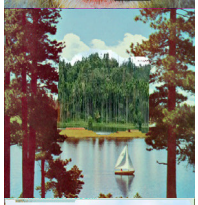
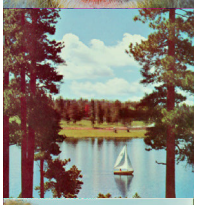
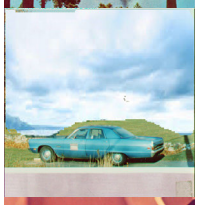
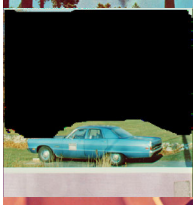
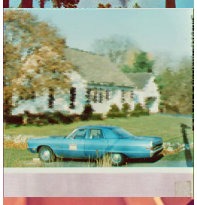
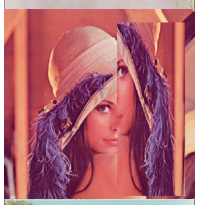
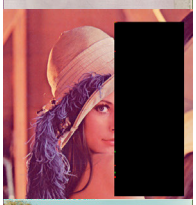
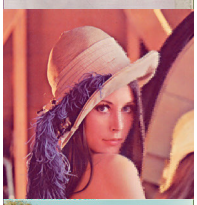
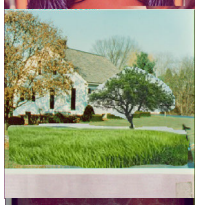
Tamper rate (%)	Attacked watermarked image	Authenticated image	Recovered image	Tamper rate (%)	Attacked watermarked image	Authenticated image	Recovered image
1				60			
5				70			
15				80			
25				85			
40				90			
50				95			

TABLE 7. Imperceptibility (PSNR, SSIM) results for different tampering rates between watermarked and recovered images.

Tampering rate (%)	PSNR, SSIM							
	Lena	Airplane	mandrill	sailboat	pepper	House	Tiffany	Splash
1	52.77, 0.9999	52.42, 0.9991	37.92, 0.9968	44.23, 0.9986	46.96, 0.9997	41.70, 0.9976	57.22, 0.9997	44.97, 0.9986
5	46.46, 0.9997	48.18, 0.9960	31.35, 0.9810	36.86, 0.9935	38.17, 0.9976	35.33, 0.9892	45.31, 0.9979	40.38, 0.9966
15	38.27, 0.9981	37.30, 0.9854	27.66, 0.9478	31.14, 0.9802	33.20, 0.9922	30.60, 0.9724	38.27, 0.9944	35.82, 0.9923
25	33.45, 0.9911	32.32, 0.9692	25.87, 0.9199	29.40, 0.9693	31.51, 0.9887	27.70, 0.9488	34.88, 0.9898	34.40, 0.9889
40	29.73, 0.9793	28.16, 0.9331	24.09, 0.8820	27.43, 0.9537	28.62, 0.9767	25.58, 0.9175	32.60, 0.9837	30.49, 0.9775
50	27.97, 0.9701	26.42, 0.9069	22.99, 0.8536	26.16, 0.9369	27.24, 0.9684	24.61, 0.9009	31.37, 0.9787	29.35, 0.9711
60	26.31,0.9582	24.96, 0.8772	21.66, 0.7996	24.37, 0.9054	25.51, 0.9552	23.33, 0.8745	30.05, 0.9722	27.46, 0.9598
70	24.74, 0.9447	23.83, 0.8440	20.43, 0.7350	22.56, 0.8650	23.39, 0.9328	22.11, 0.8408	28.52, 0.9633	25.93, 0.9489
80	23.20, 0.9296	22.54, 0.8026	19.27, 0.6609	20.88, 0.8232	21.25, 0.9062	21.10, 0.8097	26.93, 0.9507	24.55, 0.9394
85	22.17, 0.9188	21.67, 0.7770	18.59, 0.6172	19.80, 0.7901	19.77, 0.8848	20.32, 0.7855	25.90, 0.9413	23.33, 0.9290
90	20.27, 0.8971	20.16, 0.7367	17.56, 0.5725	18.11, 0.7443	17.39, 0.8519	19.03, 0.7516	24.42, 0.9259	20.59, 0.9041
95	12.48, 0.7122	15.15, 0.5542	12.19, 0.4203	11.43, 0.5314	11.05, 0.6546	13.36, 0.5579	17.11, 0.7685	11.43, 0.6850

But it does not provide significant restoration for tampering rate greater than 50%. Therefore, the block size should be selected to maintain localization accuracy and sufficient space to preserve the data. As shown in Table 5, the proposed scheme has higher PSNR (watermarked) values than the other schemes. Additionally, the significant recovery even in case of high tampering has been attained as compared to other schemes.

TABLE 8. Image authentication and self-recovery results for different types of manipulative tampering attacks.

Attacked image	Authenticated image	Tamper detection results	Recovered image	Imperceptibility results for recovered image (PSNR, SSIM)
		$N_{attacked}$ = 11574 $N_{detected}$ = 11549 TDR_{ACC} = 99.786 % Tampering rate ≅ 11.77 %		46.26, 0.9910
		$N_{attacked}$ = 53862 $N_{detected}$ = 53816 TDR_{ACC} =99.915% Tampering rate ≅ 54.79 %		22.56, 0.8250
		$N_{attacked}$ = 19263 $N_{detected}$ = 19239 TDR_{ACC} =99.875 % Tampering rate ≅ 19.60 %		31.75, 0.9822
		$N_{attacked}$ = 60290 $N_{detected}$ = 60253 TDR_{ACC} = 99.938 % Tampering rate ≅ 61.33 %		22.99, 0.8683
		$N_{attacked}$ = 32994 $N_{detected}$ = 32981 TDR_{ACC} = 99.960 % Tampering rate ≅ 33.56 %		31.19, 0.9864
		$N_{attacked}$ = 28179 $N_{detected}$ = 28168 TDR_{ACC} = 99.960 % Tampering rate ≅ 28.67 %		28.69, 0.9615
		$N_{attacked}$ = 21858 $N_{detected}$ = 21753 TDR_{ACC} = 99.520 % Tampering rate ≅ 22.24 %		32.99, 0.9881

As per the available literature of fragile watermarking, many fragile watermarking methods do not provide significant recovery of tampered parts of the image in case of higher tampering rates. Many methods report the recovery of tampered parts of the image while the tampering percentage is up to 50%. For a higher percentage of tampering, their ability to recover the image is not reported in the literature. The proposed work presents the recovery results for higher tampering percentage, which shows that the scheme can recover images effectively even when the images are highly tampered. The main reason of the same is use of 6 MSB bits for recovery procedure. The embedding strategy is design to provide large space to self-recovery information. The imperceptibility results in terms of PSNR and SSIM are

TABLE 9. Self-recovery Results (PSNR, SSIM) for different tampering rates between watermarked and recovered images.

Watermarked Images	Tampering rate (%)							
	10	20	30	40	50	60	70	80
W-1	40.94, 0.9991	35.76, 0.9955	31.96, 0.9872	29.73, 0.9793	27.98, 0.9701	26.32, 0.9582	24.75, 0.9447	23.20, 0.9296
W-2	41.57, 0.9914	33.89, 0.9770	30.64, 0.9579	28.17, 0.9331	26.42, 0.9069	24.97, 0.8772	23.83, 0.8440	22.55, 0.8026
W-3	28.84, 0.9613	26.59, 0.9324	25.20, 0.9068	24.10, 0.8820	22.99, 0.8536	21.67, 0.7996	20.44, 0.7350	19.28, 0.6609
W-4	32.54, 0.9854	30.12, 0.9745	28.83, 0.9647	27.43, 0.9537	26.17, 0.9369	24.38, 0.9054	22.57, 0.8650	20.89, 0.8232
W-5	35.22, 0.9948	32.40, 0.9908	30.34, 0.9853	28.62, 0.9767	27.25, 0.9684	25.52, 0.9552	23.40, 0.9328	21.26, 0.9062
W-6	32.26, 0.9808	29.05, 0.9615	26.72, 0.9360	25.59, 0.9175	24.62, 0.9009	23.34, 0.8745	22.11, 0.8408	21.10, 0.8097
W-7	40.88, 0.9960	36.40, 0.9924	34.00, 0.9879	32.61, 0.9837	31.38, 0.9787	30.05, 0.9722	28.53, 0.9633	26.93, 0.9507
W-8	37.20, 0.9939	34.87, 0.9905	32.22, 0.9845	30.50, 0.9775	29.36, 0.9711	27.46, 0.9598	25.94, 0.9489	24.55, 0.9394

TABLE 10. Comparison of proposed scheme with existing schemes in terms of average values for recovered images.

Tampering rate (%)	PSNR					SSIM				
	Singh [19]	Fan [20]	Tai [21]	Molina-Garcia [26]	Proposed method	Singh [19]	Fan [20]	Tai [21]	Molina-Garcia [26]	Proposed method
10	26.55	31.47	25.89	**37.34**	36.1812	0.9290	0.9731	0.9384	0.9714	**0.9878**
20	21.47	28.36	20.57	**33.98**	32.3850	0.8310	0.9502	0.8443	0.9390	**0.9768**
30	18.27	21.62	17.43	**31.28**	29.9887	0.7257	0.8875	0.7364	0.8977	**0.9638**
40	15.96	15.79	15.21	**28.47**	28.3438	0.6215	0.7230	0.6226	0.8368	**0.9504**
50	14.16	15.69	13.54	26.00	**27.0212**	0.5139	0.7202	0.5135	0.7571	**0.9358**
60	12.59	11.57	12.01	23.51	**25.4638**	0.3984	0.4249	0.3899	0.6460	**0.9128**
70	11.29	11.57	10.80	21.23	**23.9462**	0.2855	0.4249	0.2744	0.5157	**0.8843**
80	10.23	08.10	09.81	19.20	**22.4700**	0.1799	0.0094	0.1655	0.3958	**0.8528**

calculated for different tampering rates, which show that the scheme can recover images effectively even when the images are highly tampered. The authenticated and recovered images for different sizes of tampering are shown in Table 6. Table 7 presents the PSNR and SSIM values between watermarked and recovered images for the test images used in the experiment. Table 8 displays the additional authentication and self-recovery results for different manipulative tampering attacks on different watermarked images.

The simulation results show that the proposed method recovers the tampered image significantly. Table 9 presents the additional recovery results for the different tampering rates that would be used further, to compare the performance of the proposed work with the existing schemes. Although the quality of the recovered image degrades with the increase in tampering percentage, the proposed method provides better results as compared to some recently proposed methods as shown in Table 10. It is quite evident that the proposed method recovers the tampered image significantly well as compared to existing schemes. The average values of parameters (e.g. PSNR, SSIM) for the test images have been used to compare the proposed work with the existing methods. The average values of PSNR and SSIM are obtained during the self-recovery of the test images for different tampering rates.

A. PERFORMANCE ANALYSIS

The PSNR and SSIM values between host and watermarked images as have shown in Table 1, which show the superiority of the proposed scheme over [19], [20], [21] and [26]. Generally, LSB bits are modified in fragile watermarking schemes. It is obvious that the watermarking embedding capacity feature BPP (Bits per pixel) has an inverse effect on imperceptibility. The BPP values for the watermarking schemes [19], [20], [21] and [26] are 3, 2, 2 and 2 respectively. On the other hand, the BPP value for the proposed scheme is 1.5. It proves that, the proposed scheme offers least changes in host image during embedding as compared to other schemes. Therefore, the imperceptibility of the proposed scheme is higher than the existing schemes [19], [20], [21] and [26].

Although the recovery results of the proposed scheme are superior as compared to the other existing schemes, the accuracy of recovered image decreases for the images having frequent pixel variation (i.e. Mandrill, etc.) especially for low tampering rates. According to Table 10, [26] give better PSNR than the proposed scheme for low tampering rates. This degradation in PSNR for low tampering rate is because the recovery is based on blocks (i.e. 2×4 size) instead of pixels of the image. However, the SSIM results show the superiority of the proposed scheme over [26] and

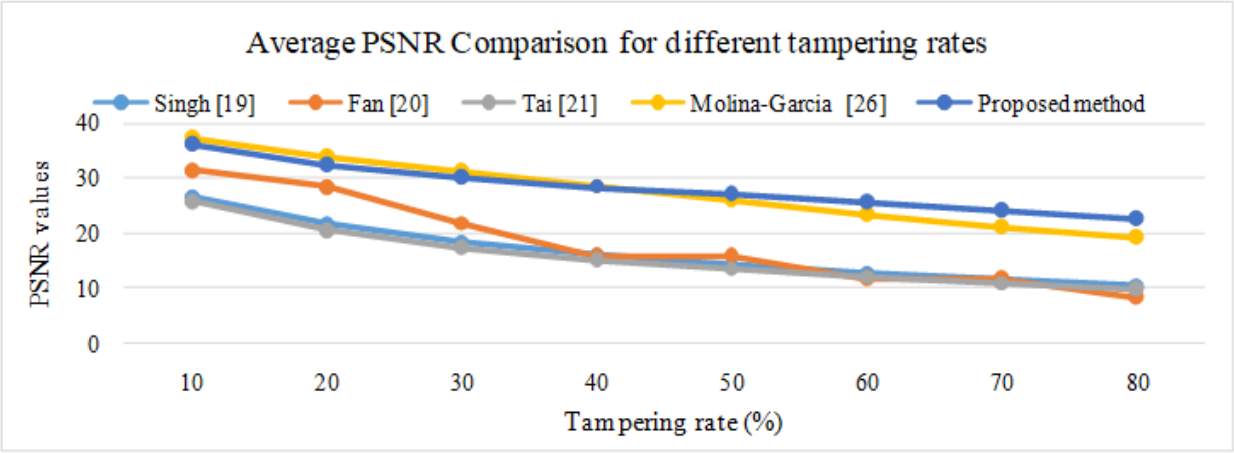


FIGURE 5. Comparative analysis of the proposed method with existing methods for Average PSNR at different tampering rates.

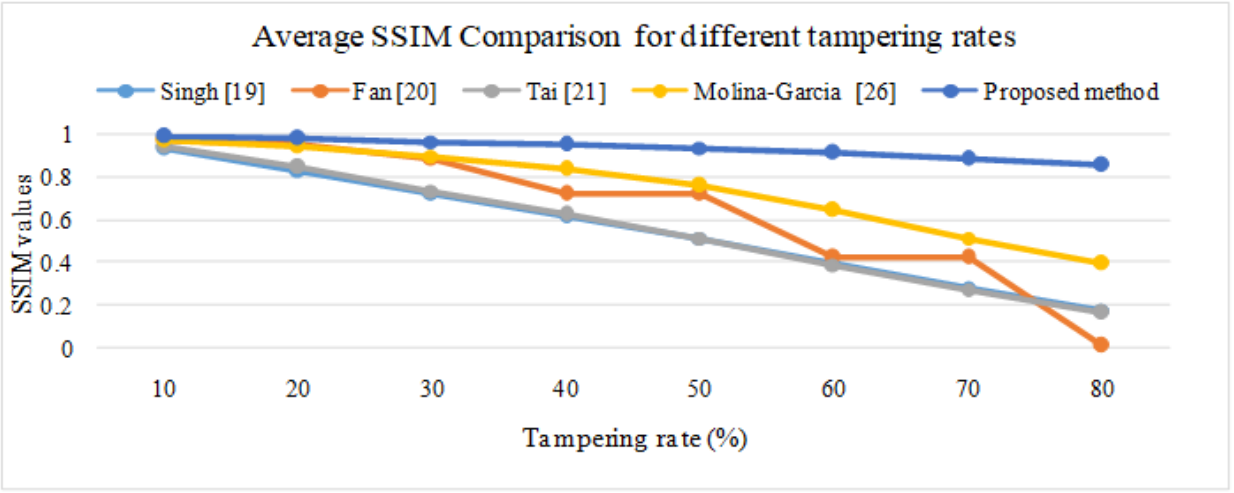


FIGURE 6. Comparative analysis of the proposed method with existing methods for Average SSIM at different tampering rates.

other schemes. SSIM compares the structural similarity, which is an improved approach to check the similarity between images because the human visual system is very much accustomed to extract the structural data from the image [31].

The block-wise division is commonly used in watermarking. Likewise, the existing schemes [20], [21] and [26] have used block-wise division to divide the host image into 8×8, 4×4 and 4×4 block sizes respectively during watermark embedding. In [20], [26] and the proposed scheme, the recovery data of the block is obtained as the 6 MSB bits of the average pixel value of the block, which could be used further to recover the block in case of tampering/forgery. Due to small block size (i.e. 2×4), the proposed scheme can preserve more recovery information than [20] and [26]. In contrast, [21] used Haar wavelet coefficients matrix for the recovery data generation which results in 28-bit recovery data for a 4×4 block size. However, the round-off error due to inverse transform reduces the performance during the recovery of the tampered blocks. The comparative results show the dominance of the proposed work over other existing works. The method provides much better performance in terms of recovery for the images having less or moderate variations in neighbor pixel values. The graphical comparison of the proposed method with the existing methods in terms of imperceptibility of recovered images is shown in Fig. 5 and Fig. 6.

V. CONCLUSION

This study offered an efficient blind fragile watermarking scheme for color images, which can detect tampering/forgery in the image and recovered the tampered part efficiently. It efficiently performed against different types of tampering attacks. A controlled random watermark as presented in this study made the method stronger against block-wise attacks. The obtained results proved that the proposed work authenticated the image and detected the tampered portion with nearly 99% accuracy. Besides, the proposed scheme offered self-recovery of the tampered regions of the image and effectively recovered highly tampered images with acceptable parametric results as compared to many existing schemes. In future, the recovery of the tampered region will

be investigated further to get improved parametric results without affecting the nature of the scheme. The image restoration against attacks like rotation, compressed sensing and other geometric attacks will also be investigated in future.

REFERENCES

[1] H. Jiang, Y. Luo, and O. Kulemeka, "Leading in the digital age: A study of how social media are transforming the work of communication professionals," *Telematics Informat.*, vol. 33, no. 2, pp. 493–499, May 2016.
[2] J. A. Redi, W. Taktak, and J.-L. Dugelay, "Digital image forensics: A booklet for beginners," *Multimedia Tools Appl.*, vol. 51, no. 1, pp. 133–162, Jan. 2011.
[3] P. P. Dang and P. M. Chau, "Image encryption for secure Internet multimedia applications," *IEEE Trans. Consum. Electron.*, vol. 46, no. 3, pp. 395–403, Aug. 2000.
[4] P. Korus, "Digital image integrity–a survey of protection and verification techniques," *Digit. Signal Process.*, vol. 71, pp. 1–26, Dec. 2017.
[5] X. Zhang and S. Wang, "Fragile watermarking with error-free restoration capability," *IEEE Trans. Multimedia*, vol. 10, no. 8, pp. 1490–1499, Dec. 2008.
[6] P. Singh and R. S. Chadha, "A survey of digital watermarking techniques, applications and attacks," *Int. J. Eng. Innov. Technol.*, vol. 2, no. 9, pp. 165–175, Mar. 2013.
[7] N. Nikolaidis and I. Pitas, "Digital image watermarking: An overview," in *Proc. IEEE Int. Conf. Multimedia Comput. Syst.*, vol. 1, Jun. 1999, pp. 1–6.
[8] G. C. Langelaar, I. Setyawan, and R. L. Lagendijk, "Watermarking digital image and video data. A state-of-the-art overview," *IEEE Signal Process. Mag.*, vol. 17, no. 5, pp. 20–46, Sep. 2000.
[9] L.-Y. Hsu and H.-T. Hu, "A reinforced blind color image watermarking scheme based on schur decomposition," *IEEE Access*, vol. 7, pp. 107438–107452, 2019.
[10] P. Wah Wong and N. Memon, "Secret and public key image watermarking schemes for image authentication and ownership verification," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1593–1601, Oct. 2001.
[11] S. Voloshynovskiy, S. Pereira, T. Pun, J. Eggers, and J. Su, "Attacks on digital watermarks: Classification, estimation based attacks, and benchmarks," *IEEE Commun. Mag.*, vol. 39, no. 8, pp. 118–126, Aug. 2001.
[12] X. Qi and X. Xin, "A quantization-based semi-fragile watermarking scheme for image content authentication," *J. Vis. Commun. Image Represent.*, vol. 22, no. 2, pp. 187–200, Feb. 2011.
[13] P.-C. Su, H.-J. M. Wang, and C.-C. J. Kuo, "An integrated approach to image watermarking and JPEG-2000 compression," *J. VLSI Signal Process. Syst. Signal, Image Video Technol.*, vol. 27, pp. 35–53, Feb. 2001.
[14] H. Tao, L. Chongmin, J. Mohamad Zain, and A. N. Abdalla, "Robust image watermarking theories and techniques: A review," *J. Appl. Res. Technol.*, vol. 12, no. 1, pp. 122–138, Feb. 2014.
[15] S.-J. Lee and S.-H. Jung, "A survey of watermarking techniques applied to multimedia," in *Proc. IEEE Int. Symp. Ind. Electron.*, Jun. 2001, pp. 272–277.
[16] U. H. Panchal and R. Srivastava, "A comprehensive survey on digital image watermarking techniques," in *Proc. 5th Int. Conf. Commun. Syst. Netw. Technol.*, Gwalior, India, Apr. 2015, pp. 591–595.
[17] X. Zhang, S. Wang, Z. Qian, and G. Feng, "Reference sharing mechanism for watermark self-embedding," *IEEE Trans. Image Process.*, vol. 20, no. 2, pp. 485–495, Feb. 2011.
[18] H. He, F. Chen, H.-M. Tai, T. Kalker, and J. Zhang, "Performance analysis of a Block-Neighborhood-Based self-recovery fragile watermarking scheme," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 1, pp. 185–196, Feb. 2012.
[19] D. Singh and S. K. Singh, "Effective self-embedding watermarking scheme for image tampered detection and localization with recovery capability," *J. Vis. Commun. Image Represent.*, vol. 38, pp. 775–789, Jul. 2016.
[20] M. Fan and H. Wang, "An enhanced fragile watermarking scheme to digital image protection and self-recovery," *Signal Process., Image Commun.*, vol. 66, pp. 19–29, Aug. 2018.
[21] W.-L. Tai and Z.-J. Liao, "Image self-recovery with watermark self-embedding," *Signal Process., Image Commun.*, vol. 65, pp. 11–25, Jul. 2018.
[22] C. Qin, P. Ji, C.-C. Chang, J. Dong, and X. Sun, "Non-uniform watermark sharing based on optimal iterative BTC for image tampering recovery," *IEEE MultimediaMag.*, vol. 25, no. 3, pp. 36–48, Jul. 2018.
[23] X. L. Liu, C. C. Lin, and S. M. Yuan, "Blind dual watermarking for color images' authentication and copyright protection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 5, pp. 1047–1055, May 2018.
[24] N. N. Hurrah, S. A. Parah, N. A. Loan, J. A. Sheikh, M. Elhoseny, and K. Muhammad, "Dual watermarking framework for privacy protection and content authentication of multimedia," *Future Gener. Comput. Syst.*, vol. 94, pp. 654–673, May 2019.
[25] P. Pal, B. Jana, and J. Bhaumik, "Watermarking scheme using local binary pattern for image authentication and tamper detection through dual image," *Secur. Privacy*, vol. 2, no. 2, p. e59, Feb. 2019, doi: 10.1002/spy2.59.
[26] J. Molina-Garcia, B. P. Garcia-Salgado, V. Ponomaryov, R. Reyes-Reyes, S. Sadovnychiy, and C. Cruz-Ramos, "An effective fragile watermarking scheme for color image tampering detection and self-recovery," *Signal Process., Image Commun.*, vol. 81, Feb. 2020, Art. no. 115725, doi: 10.1016/j.image.2019.115725.
[27] M. Matsumoto and T. Nishimura, "Mersenne twister: A 623-dimensionally equidistributed uniform pseudo-random number generator," *ACM Trans. Model. Comput. Simul.*, vol. 8, no. 1, pp. 3–30, Jan. 1998.
[28] A. Hore and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *Proc. 20th Int. Conf. Pattern Recognit.*, Istanbul, Turkey, Aug. 2010, pp. 2366–2369.
[29] H. Huang, W. Guo, and Y. Zhang, "Detection of copy-move forgery in digital images using SIFT algorithm," in *Proc. IEEE Pacific-Asia Workshop Comput. Intell. Ind. Appl.*, Wuhan, China, Dec. 2008, pp. 272–276.
[30] P. Kakar and N. Sudha, "Exposing postprocessed copy–paste forgeries through transform-invariant features," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 1018–1028, Jun. 2012.
[31] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

RISHI SINHAL (Graduate Student Member, IEEE) received the B.E. degree in electronics and communication engineering from the Jabalpur Engineering College, Jabalpur, India, and the M.E. degree in digital techniques and instrumentation from SGSITS, Indore, India. He is currently pursuing the Ph.D. degree in electronics and communication engineering with the PDPM Indian Institute of Information Technology, Design and Manufacturing Jabalpur, India. His research interests include image watermarking, image processing, signal and image processing applications, and machine learning techniques.

IRSHAD AHMAD ANSARI (Member, IEEE) received the Ph.D. degree from IIT Roorkee, with MHRD Teaching Assistantship. He subsequently joined the Gwangju Institute of Science and Technology, South Korea, as a Postdoctoral Fellow. He is currently working as an Assistant Professor at the Discipline of Electronics and Communication Engineering, PDPM Indian Institute of Information and Technology Design and Manufacturing Jabalpur, India. His major research interests include image processing, signal processing, soft computing, data classification, and brain–computer interface. He has authored more than 30 research papers in various reputed international journals/conferences of publishers like IEEE, Elsevier, Springer, and so on. He also serves as an active and potential technical reviewer for various journals of repute.

CHANG WOOK AHN (Member, IEEE) received the Ph.D. degree from the Department of Information and Communications, Gwangju Institute of Science and Technology (GIST), Gwangju, South Korea. He worked as a Professor at Sungkyunkwan University (SKKU). He is currently working as a Professor at the School of Electrical Engineering and Computer Science, GIST. He is the Director of the Meta-Evolutionary Machine Intelligence (MEMI) Laboratory, GIST. His research interests include genetic algorithms/programming, multiobjective optimization, evolutionary neural networks, and quantum machine learning.