A DWT-based Digital Watermarking Scheme for Image Tamper Detection, Localization, and Restoration

Sukalyan Som, Sarbani Palit, Kashinath Dey, Dipabali Sarkar, Jayeeta Sarkar and Kheyali Sarkar

Abstract The provision of image tamper detection, localization and restoration forms an important requirement for modern multimedia and communication systems. A discrete wavelet transform (DWT)-based watermarking scheme for this purpose is proposed in this communication. In our scheme, the original image is first partitioned into blocks of size 2×2 in which a 1D DWT is applied to produce a watermark which is embedded in four disjoint partitions of the image to enhance the chance of restoration of the image from different cropping attack-based tampers. The validity and superiority of the proposed scheme is verified through extensive simulations using different images of two extensively used image databases.

Keywords Discrete wavelet transform (DWT) · Least significant bits (LSBs) · Peak signal-to-noise ratio (PSNR) · Mean squared error (MSE) · Structural SIMilarity (SSIM) index

S. Som (✉) · J. Sarkar · K. Sarkar
Department of Computer Science, Barrackpore Rastraguru Surendranath College, Barrackpore, West Bengal, India
e-mail: sukalyan.s@gmail.com

J. Sarkar
e-mail: sarkar.jayeeta9@gmail.com

K. Sarkar
e-mail: chelsea.kheyali9@gmail.com

S. Palit · D. Sarkar
CVPR Unit, Indian Statistical Institute, Kolkata, West Bengal, India
e-mail: sarbanip@isical.ac.in

D. Sarkar
e-mail: mampisarkar333@gmail.com

K. Dey
Department of Computer Science and Engineering, University of Calcutta, 92, APC Road, Kolkata 700009, West Bengal, India
e-mail: kndey55@gmail.com

R. Chaki et al. (eds.), *Applied Computation and Security Systems*, Advances in Intelligent Systems and Computing 305, DOI 10.1007/978-81-322-1988-0_2

1 Introduction

Tampering of digital media and its detection has been an interesting problem since long time. Its importance has increased with the stepping up of the use of digital media on the Internet. The volume of data transmission, especially that of images and videos, has gone up exponentially and has naturally drawn the interest of many including, unfortunately, fraudulent persons who would tamper with the transmitted data to suit their purpose. The detection of tampering followed by restoration of the original image is hence an important task. Most of the research carried out so far has been of tamper detection, while more recent work includes recovery of the image as well.

A number of digital watermarking schemes have been reported during the past decade for different purposes and considerations. In [1], an image tamper detection and recovery system has been developed based on the discrete wavelet transform (DWT) technique where some information has been extracted as the eigenvalue of the image and is embedded in the middle-frequency band of the frequency domain. Such embedding has been used for tamper detection and localization. In [2], a novel fragile watermarking scheme based on chaotic system for image authentication or tamper proofing is proposed. The watermark is generated by using pixel values as input values of a chaotic system, and a secret key controls a set of parameters of the chaotic system. A quantization function is introduced to embed and detect watermarks. This method can effectively detect minor alteration in a watermarked image. In [3], a tamper detection and retrieval scheme has been proposed. Special characteristic values of the low-frequency sub-band are embedded in the middle-frequency sub-bands. The embedded data with a digital signature and a public key are used to prove the authenticity of the image. Recovery with visually acceptable quality has also been achieved. In [4], the watermark of a particular image is generated from both frequency domain and spatial domain. The number of encoding stages of each DWT coefficient during the multistage encoding is taken as frequency watermark, and the mean values of blocks are stored as spatial watermark. The watermark is embedded into SPIHT encoded list of significant pixels (LSP) bit stream. By comparing the embedded watermark and the corresponding message extracted from decoded image, authentication is ensured. In [5], the semi-fragile watermark is designed from low-frequency band of wavelet-transformed image and is embedded into the high-frequency band by the human visual system (HVS). The robustness for mild modification such as JPEG compression and channel additive white Gaussian noise (AWGN) and fragility to malicious attack are analyzed. In [6], the proposed scheme extracts content-based image features from the approximation sub-band in the wavelet domain to generate two complementary watermarks. An edge-based watermark sequence is generated to detect any changes after manipulations. A content-based watermark is also generated to localize tampered regions. Both watermarks are embedded into the high-frequency wavelet domain to ensure the watermark invisibility. In [7], the original image is divided into two regions: region of interest (ROI), which is important region that

requires protection against malicious modification, and region of embedding (ROE), which is the rest of the image where watermark sequence is embedded. In [8], dual visual watermarks using DWT and singular value decomposition (SVD) are presented. One is color image the same as original image, and the other is ownership watermark which is grayscale image. Both of them are embedded into original image using DWT-SVD to prove robustness. For recovery signal embedding, luminance signal and chrominance signal of original image were embedded into surplus chrominance space of original image using matrix transpose replacement embedding method. In [9, 10], two watermarks are used, generated from the low-frequency band and embedded into the high-frequency bands, one for detecting the intentional content modification and indicating the modified location and another for recovering the image. In [11], a multipurpose image watermarking method based on the wavelet transform is proposed for content authentication and recovery of the tampered regions where the original image is first divided into non-overlapping blocks and each block is transformed into the wavelet domain. The image features are subsequently extracted from the lowest frequency coefficients of each block as the first embedded watermark. Next, the whole image is decomposed into the two-level wavelet transform, and the orientation adjustment is calculated based on the wavelet coefficients in the middle-frequency sub-bands for image authentication. In addition, a logo watermark is embedded into the given middle-frequency sub-bands.

The rest of the paper is organized as follows. In Sect. 2, a brief introduction to DWT using Haar wavelet is given. In Sect. 3, the proposed scheme is presented wherein watermark generation, watermark embedding, and watermark extraction for the purpose of image tamper detection, localization, and recovery are explained. Section 4 demonstrates the experimental results with conclusions being drawn in Sect. 5.

2 Background

2.1 Discrete Wavelet Transform

The single-level 2D DWT decomposes an input image into four components, namely LL, LH, HL, and HH where the first letter corresponds to applying either a low-pass or a high-pass frequency operation to the rows and the second letter refers to the filter applied to the columns. The lowest frequency sub-band LL consists of the approximation coefficients of the original image. The remaining three frequency sub-bands consist of the detail parts and give the vertical high (LH), horizontal high (HL), and high (HH) frequencies. Figure 1 demonstrates single-level 2D DWT. For an one-level decomposition, the discrete 2D wavelet transform of the image function $f(x, y)$ can be written as follows:

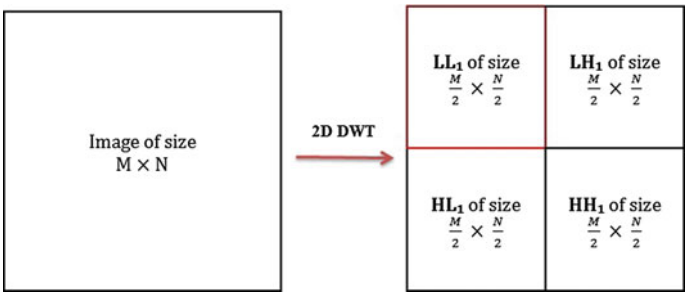


Fig. 1 Discrete wavelet transform

$$\begin{aligned}
\mathrm{LL} &= [(f(x,y) \times \phi - x\phi - y)(2n, 2m)]_{(n,m)\in Z^2} \\
\mathrm{LH} &= [(f(x,y) \times \phi - x\psi - y)(2n, 2m)]_{(n,m)\in Z^2} \\
\mathrm{HL} &= [(f(x,y) \times \psi - x\phi - y)(2n, 2m)]_{(n,m)\in Z^2} \\
\mathrm{HH} &= [(f(x,y) \times \psi - x\psi - y)(2n, 2m)]_{(n,m)\in Z^2}
\end{aligned}$$

where $\phi(t)$ is a low-pass scaling function and $\psi(t)$ is the associated band-pass wavelet function. For computational simplicity, we have performed DWT using Haar wavelet.

3 Proposed Scheme

The proposed method has three distinct phases. Firstly, a watermark is generated from the image itself which is fragile to content modification as well as robust to common image processing after a preparation for doing so. Secondly, the generated watermark is embedded in the image. Finally, the watermark is extracted from the image (the one that has gone several degradations due to cropping attacks and/or noise attacks) to detect and localize tamper and recover the image as close as possible to the original one.

3.1 Watermark Preparation

A block mapping sequence is used to scramble watermark information. A 1D transformation algorithm, found in [12], shown in Eq. (1) is used to obtain a one-to-one mapping sequence where $X, X'(\in [0, N-1])$ the block number, k(a prime and $\in Z - \{\text{factors of } N\}$) is a secret key, and $N(\in Z - \{0\})$ is the total number of blocks in the image of size $N = 2^n \times 2^n$, $n \geq 2$, and $n \in N$.

$$X' = [f(x) = (k \times X) \bmod N] + 1 \tag{1}$$

A lookup table is constructed using the following algorithm to record the mapping address of each block in the image.

3.1.1 Block Mapping Address Generation Algorithm

1. Divide the image into non-overlapping blocks of 2 × 2 pixels.
2. Assign a unique nonnegative integer $X \in \{0, 1, 2, \ldots N-1\}$ to each block from top left in row major order, $N = 2^{n-1} \times 2^{n-1}$.
3. Choose a prime number $k \in [1, N-1]$.
4. For each block number X, obtain X' and its mapping block by Eq. (1). All the X's construct the lookup table.

A push-aside operation is used to modify the lookup table. The watermarks of the left half of the image are concentrated in the right half region of the image, and the watermarks of the right half of the image are concentrated in the left half region of the image. We simply push right the columns which originally belong to the left half and push left the columns which originally belong to the right half and thus result in a modified lookup table.

As an illustration, an image of size 8 × 8 is considered as the original image. The original image along with its corresponding block index matrix, lookup table generated using Eq. (1), and modified lookup table after push-aside operation is shown in Fig. 2.

3.2 Watermark Generation

Step 1: Decompose each 2 × 2 sized block by the DWT decomposition yielding from each block the approximation coefficient matrix LL_1 and the detail matrices HL_1, LH_1, and HH_1.

Step 2: The watermark is generated from the coefficient of the LL_1 sub-band of each decomposed block. As LL_1 wavelet coefficients may be beyond the recovery scope, its value must be adjusted. Therefore, the coefficients, after computation, are modified subsequently such that its value falls within the recovery range, as done in [5].

Step 3: The original image is divided horizontally and vertically into four equal parts. Let blocks A, B, C, and D be located at those four parts, respectively, such that C is situated at the opposite angle of A and D is situated at the opposite angle of B. Partner blocks of part A are located at the same position of part C and vice versa. Partner blocks of part B are located at the same position of part D and vice versa.

Step 4: The representative information of block A is constructed by extracting the five most significant bits (MSBs) of LL_1 sub-band coefficient of block A

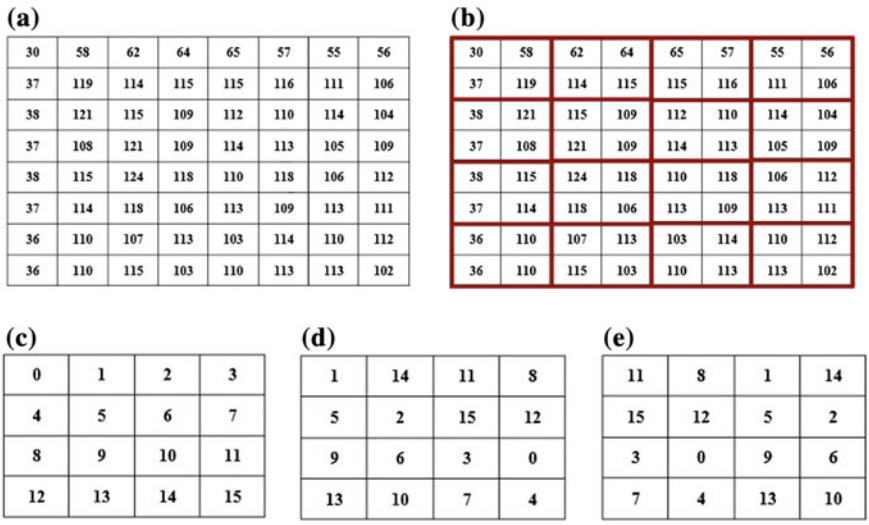


Fig. 2 **a** The original image matrix; **b** the original image matrix subdivided into 2 × 2 non-overlapping blocks; **c** the original block matrix; **d** the lookup table; and **e** the modified lookup table after push-aside operation

and is then combined with (1) the representative information of block C and (2) the in-block parity-check bits and its complementary bit *p* and *v,* respectively, to construct the joint 12-bit watermark for blocks A and C. Similarly, the representative information of block B is used to construct the joint 12-bit watermark for blocks B and D.

The watermark generation technique is illustrated in Figs. 3 and 4.

3.3 Watermark Embedding

Two mapping blocks are needed to embed the joint 12-bit watermark of block A (or B) and its partner blocks C (or D). The lookup table helps find these mapping blocks. The watermark is embedded into the three LSBs of each pixel of a block. Suppose blocks $\overline{A}$ and $\overline{C}$ (or $\overline{B}$ and $\overline{D}$) are the two mapping blocks which are going to be used to embed the 12-bit watermark resulted from blocks A and C (or B and D). Both blocks $\overline{A}$ and $\overline{C}$ contain the same 12-bit watermark and the same embedding sequence in the corresponding locations. That is to say, for each block of size 2 × 2 pixels in the image, we have two copies of its representative information hidden somewhere in the image. Therefore, if one copy is tampered by any chance, we have two chances to recover this block from the other copy.

Figures 5 and 6 demonstrate the watermark embedding technique.

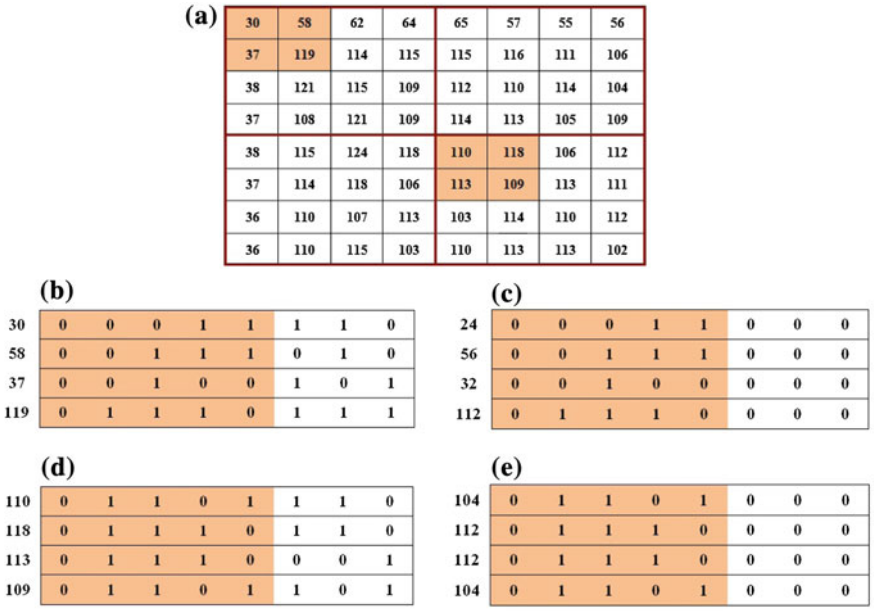


Fig. 3 **a** First two partner blocks (block 0 and block 10) in the original image matrix; **b** binary equivalent of each of the four pixels of block 0; **c** modified pixel values of block 0 after replacing three LSBs with 0s; **d** binary equivalent of each of the four pixels of block 10; and **e** modified pixel values of block 10 after replacing three LSBs with 0s

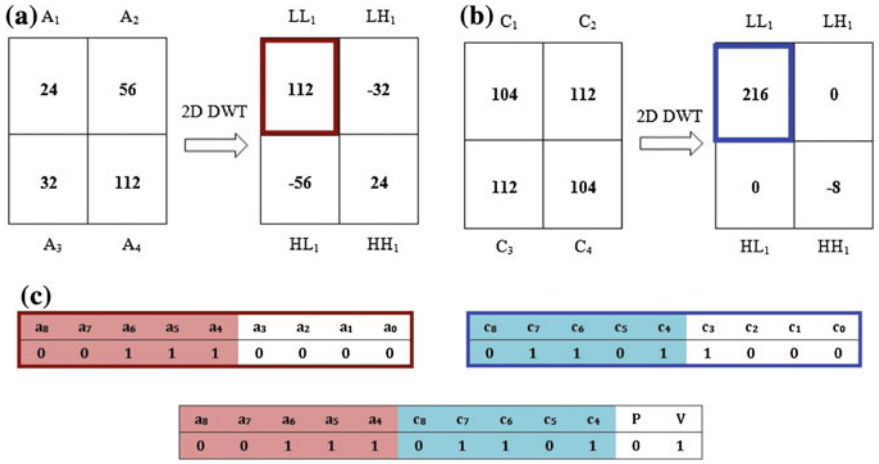


Fig. 4 **a** and **b** Application of 2D DWT using Haar wavelets into block 0 and block 10, respectively, resulting in the approximation coefficient matrix LL_1 and detail matrices LH_1, HL_1, and HH_1 and **c** the 12-bit watermark generated from the five MSBs of the LL_1 sub-band coefficient of block 0 and block 10 followed by a in-block parity-check bit P and its complement V

(a)

Block No.	Mapping Block No.
0	11
10	9

(b)

11	8	1	14
15	12	5	2
3	0	9	6
7	4	13	10

(c)

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

(d)

30	58	62	64	65	57	55	56
37	119	114	115	115	116	111	106
38	121	115	109	112	110	114	104
37	108	121	109	114	113	105	109
38	115	124	118	110	118	106	112
37	114	118	106	113	109	113	111
36	110	107	113	103	114	110	112
36	110	115	103	110	113	113	102

Fig. 5 **a** Mapping blocks block 11 and block 9 of block 0 and block 10, respectively, found from the modified lookup table; **b** mapping blocks highlighted in the modified lookup table; **c** mapping blocks highlighted in the original block matrix; and **d** pixels of mapping blocks highlighted in the original image matrix

(a)

	$\bar{a}_7$	$\bar{a}_6$	$\bar{a}_5$	$\bar{a}_4$	$\bar{a}_3$	$\bar{a}_2$	$\bar{a}_1$	$\bar{a}_0$
106	0	1	1	0	1	0	1	0
112	0	1	1	1	0	0	0	0
113	0	1	1	1	0	0	0	1
111	0	1	1	0	1	1	1	1

	$\bar{c}_7$	$\bar{c}_6$	$\bar{c}_5$	$\bar{c}_4$	$\bar{c}_3$	$\bar{c}_2$	$\bar{c}_1$	$\bar{c}_0$
124	0	1	1	1	1	1	0	0
118	0	1	1	1	0	1	1	0
118	0	1	1	1	0	1	1	0
106	0	1	1	0	1	0	1	0

(b)

$\bar{a}_7$	$\bar{a}_6$	$\bar{a}_5$	$\bar{a}_4$	$\bar{a}_3$	$\bar{a}_2$	$\bar{a}_1$	$\bar{a}_0$
0	1	1	0	1	a_8	a_7	a_6
0	1	1	1	0	a_5	a_4	c_8
0	1	1	1	0	c_7	c_6	c_5
0	1	1	0	1	c_4	P	V

$\bar{c}_7$	$\bar{c}_6$	$\bar{c}_5$	$\bar{c}_4$	$\bar{c}_3$	$\bar{c}_2$	$\bar{c}_1$	$\bar{c}_0$
0	1	1	1	1	a_8	a_7	a_6
0	1	1	1	0	a_5	a_4	c_8
0	1	1	1	0	c_7	c_6	c_5
0	1	1	0	1	c_4	P	V

(c)

	$\bar{a}_7$	$\bar{a}_6$	$\bar{a}_5$	$\bar{a}_4$	$\bar{a}_3$	$\bar{a}_2$	$\bar{a}_1$	$\bar{a}_0$
105	0	1	1	0	1	0	0	1
118	0	1	1	1	0	1	1	0
118	0	1	1	1	0	1	1	0
109	0	1	1	0	1	1	0	1

	$\bar{c}_7$	$\bar{c}_6$	$\bar{c}_5$	$\bar{c}_4$	$\bar{c}_3$	$\bar{c}_2$	$\bar{c}_1$	$\bar{c}_0$
121	0	1	1	1	1	0	0	1
118	0	1	1	1	0	1	1	0
118	0	1	1	1	0	1	1	0
109	0	1	1	0	1	1	0	1

(d)

30	58	62	64	65	57	55	56
37	119	114	115	115	116	111	106
38	121	115	109	112	110	114	104
37	108	121	109	114	113	105	109
38	115	121	118	110	118	105	118
37	114	118	109	113	109	118	109
36	110	107	113	103	114	110	112
36	110	115	103	110	113	113	102

Fig. 6 **a** Binary representation of each of the four pixels of the mapping blocks—block 11 and block 9; **b** embedding of the same 12-bit watermark into block 11 and block 9; **c** modified block 11 and block 9 after watermark embedding; and **d** modified block 11 and block 9 in the original image matrix

3.4 Watermark Extraction: Tamper Detection, Localization, and Restoration

The watermarked image is tampered with different cropping attacks and covering and replacement attacks. Figure 7 represents the watermarked image of Fig. 6e cropped 25 % from center.

Tamper detection and localization A three-level hierarchical tamper detection and localization algorithm has been employed as proposed in [12].

Level 1 detection: For each non-overlapping block B of size 2 × 2,

1. Retrieve the 12-bit watermark information from the block.
2. Get the parity-check bits p and v, respectively, from the 11th and 12th bits of the retrieved watermark.
3. Perform exclusive-OR operation on the 10 MSBs of the 12-bit watermark, denoted by p'.
4. If $p = p'$ and $p \neq v$, mark block B valid; otherwise, mark it invalid.

Figure 8 demonstrates the level 1 tamper detection method.

Level 2 detection: For each block B marked valid after level 1 detection, check four triples (N, NE, E), (E, SE, S), (S, SW, W), and (W, NW, N) of the 3 × 3 neighborhood of block B. If at least one triple has all of its blocks marked invalid, mark block B invalid.

Level 3 detection: For each block B marked valid after level 2 detection, if at least five of the 3 × 3 neighboring blocks of block B are marked invalid, mark block B invalid.

Recovery of invalid blocks After the tamper detection process, all blocks in the image are marked either valid or invalid. Those invalid blocks need only to be recovered. A two-stage recovery scheme is applied for tamper recovery as follows:

Stage 1 recovery: For each non-overlapping block B of size 2 × 2 pixels which is marked invalid,

1. Find the mapping block of B from the lookup table, denoted by $\overline{B}$

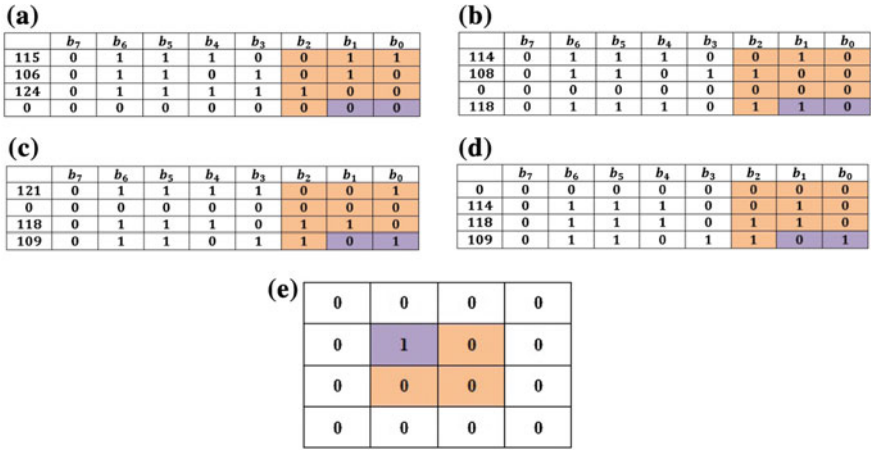
(a)

26	58	58	68	67	58	50	60
39	114	116	117	118	117	108	109
35	122	115	106	114	108	115	106
38	109	124	0	0	118	108	105
34	116	121	0	0	114	105	118
38	118	118	109	118	109	118	109
35	106	106	114	98	114	106	114
38	109	118	102	111	114	118	102

(b)

26	58	58	68	67	58	50	60
39	114	116	117	118	117	108	109
35	122	115	106	114	108	115	106
38	109	124	0	0	118	108	105
34	116	121	0	0	114	105	118
38	118	118	109	118	109	118	109
35	106	106	114	98	114	106	114
38	109	118	102	111	114	118	102

Fig. 7 **a** Tampered image after cropping 25 % from the center of the watermarked image and **b** image in (**a**) with blocks highlighted



(a)	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
115	0	1	1	1	0	0	1	1
106	0	1	1	0	1	0	1	0
124	0	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0

(b)	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
114	0	1	1	1	0	0	1	0
108	0	1	1	0	1	1	0	0
0	0	0	0	0	0	0	0	0
118	0	1	1	1	0	1	1	0

(c)	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
121	0	1	1	1	1	0	0	1
0	0	0	0	0	0	0	0	0
118	0	1	1	1	0	1	1	0
109	0	1	1	0	1	1	0	1

(d)	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
0	0	0	0	0	0	0	0	0
114	0	1	1	1	0	0	1	0
118	0	1	1	1	0	1	1	0
109	0	1	1	0	1	1	0	1

(e)				
	0	0	0	0
	0	1	0	0
	0	0	0	0
	0	0	0	0

Fig. 8 Level 1 tamper detection and localization: **a** four pixels of block 5 with their binary equivalents; **b** four pixels of block 6 with their binary equivalents; **c** four pixels of block 9 with their binary equivalents; **d** four pixels of block 10 with their binary equivalents; and **e** localization of tampered block(s) after level 1 detection

2. If $\overline{B}$ is valid, then $\overline{B}$ is the candidate block, go to 5.
3. Find the mapping block of B′s partner block, denoted by $\overline{\overline{B}}$.
4. If $\overline{\overline{B}}$ is valid, then $\overline{\overline{B}}$ is the candidate block; otherwise stop, leave block B alone.
5. Retrieve the 12-bit watermark information from the candidate block.
6. If block B is located in the upper half of the image, the 5-bit representative information of block B starts from the first bit (the leftmost bit) of the 12-bit watermark; otherwise, it starts from the sixth bit.
7. Pad four 0s to the end of the 5-bit representative information to form a new 9-bit coefficient.
8. Perform the inverse DWT operation based on this coefficient as the approximation coefficient which generates a new block of size 2×2.
9. Replace block B with this new block and mark block B as valid.

The method for stage 1 recovery is shown in Fig. 9.

Stage 2 recovery: Recover the remaining invalid blocks after stage 1 recovery from the neighboring pixels surrounding them. Corresponding to a central block B being processed, the 3×3 neighboring blocks can be found as directional triples (N, NE, E), (E, SE, S), (S, SW, W), and (W, NW, N) where each of the neighboring blocks being denoted as N_1–N_8 from NW to W in a clockwise manner. After the two-stage recovery process, lost blocks are reconciled by interpolating pixel values.

Figure 10 presents the reconstructed image of Fig. 7 after stage 2 recovery.

(a)

Tampered Block No.	Mapping Block No.
5	12

(b)

	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
35	0	0	1	0	0	0	1	1
106	0	1	1	0	1	0	1	0
38	0	0	1	0	0	1	1	0
109	0	1	1	0	1	1	0	1

(c)

	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0
208	0	1	1	0	1	0	0	0	0

(d)

106	104
104	104

(e)

26	58	58	68	67	58	50	60
39	114	116	117	118	117	108	109
35	122	106	104	111	111	115	106
38	109	104	104	0	111	108	105
34	116	109	0	0	109	105	118
38	118	109	109	109	109	118	109
35	106	106	114	98	114	106	114
38	109	118	102	111	114	118	102

Fig. 9 Stage 1 recovery: **a** mapping block of the detected tampered block; **b** four pixels of the mapping block (block 12) with 5-bit information of block 5 embedded as watermark; **c** 5-bit information of block 5 padded with four 0s forming 9-bit approximation of block 5; **d** reconstructed block resulting from 2D inverse DWT on (**c**); and **e** recovered image after stage 1 recovery

(a)

26	58	58	68	67	58	50	60
39	114	116	117	118	117	108	109
35	122	106	104	111	111	115	106
38	109	104	104	111	111	108	105
34	116	109	109	109	109	105	118
38	118	109	109	109	109	118	109
35	106	106	114	98	114	106	114
38	109	118	102	111	114	118	102

(b)

26	58	58	68	67	58	50	60
39	114	116	117	118	117	108	109
35	122	106	104	111	111	115	106
38	109	104	104	111	111	108	105
34	116	109	109	109	109	105	118
38	118	109	109	109	109	118	109
35	106	106	114	98	114	106	114
38	109	118	102	111	114	118	102

Fig. 10 Stage 2 recovery: **a** and **b** the recovered image after reconciling the missing blocks by interpolating pixel values

4 Experimental Results

The performance and feasibility of the proposed scheme is examined through extensive tests carried out over USC-SIPI [13] and CSIQ [14] image databases which are collections of digitized images available and maintained by University of Southern California and School of Electrical and Computer Engineering,

Oklahoma State University, respectively. The images are chosen to prove the efficacy of the proposed scheme over various characteristics such as smooth areas, edges, textures, curvature, and regular and irregular geometric objects. The proposed scheme and the existing state of the art, considered for comparison, have been implemented using MATLAB 7.10.0.4 (R2010a) on a system running on Windows 7 (32 bit) with Intel Core i5 CPU and 4-GB DDR3 RAM.

The proposed scheme was examined against cropping attacks of different sizes. The performance of the proposed method is measured by the peak signal-to-noise ratio (PSNR) and Structural SIMilarity (SSIM) index [15].

The PSNR of a given image is the ratio of the mean square difference of two images to the maximum mean squared difference that can exist between any two images. It is expressed as a decibel value. An image with a PSNR value of 30 dB or more is widely accepted as an image of good quality. SSIM measures the similarity/dissimilarity between two images. For a watermarked image, greater value of PSNR and SSIM close to unity is expected.

Let $I_1(i, j)$ and $I_2(i, j)$ be the gray level of the pixels at the ith row and jth column of two images of size $H \times W$, respectively. The MSE between these two images is defined in Eq. (2), and PSNR is defined in Eq. (3).

$$\mathrm{MSE} = \frac{1}{H \cdot W} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} |I_1(i,j) - I_2(i,j)|^2 \tag{2}$$

$$\mathrm{PSNR} = 20 * \log_{10}\left(\frac{255}{\mathrm{sqrt(MSE)}}\right) \tag{3}$$

The ***SSIM*** index between two images I_1 and I_2 as described in [15] is computed using Eq. (4):

$$\mathrm{SSIM}(I_1, I_2) = \frac{(2\mu_{I_1}\mu_{I_2} + C_1)(2\sigma_{I_1 I_2} + C_2)}{(\mu_{I_1}{}^2 + \mu_{I_2}{}^2 + C_1)(\sigma_{I_1}{}^2 + \sigma_{I_2}{}^2 + C_2)} \tag{4}$$

where μ, σ, and σ^2 denote average, variance, and covariance, respectively, and C_1 and C_2 are constants as described in detail in [15].

4.1 Imperceptibility of Watermark

Imperceptible watermarks are invisible to naked eyes. If the embedded watermark is imperceptible, human eye cannot discriminate between the original image and its watermarked version. In the proposed scheme, the imperceptibility of the watermark has been examined for a wide variety of images in terms of PSNR and SSIM. For the watermarked images, greater value of PSNR (well above 35) and SSIM close to unity justify the imperceptibility of the watermark. A sample image

Fig. 11 **a** Original image of Lena; **b** watermarked image of (**a**)

Table 1 Comparison of PSNR and SSIM of watermarked images

Image name	Size	Ref. [12]		Proposed	
		PSNR (in dB)	SSIM	PSNR (in dB)	SSIM
Lena	512 × 512	41.44	0.93	41.44	0.93
Peppers	512 × 512	41.39	0.93	41.39	0.93
Baboon	512 × 512	41.30	0.98	41.31	0.98
Boat	512 × 512	41.35	0.95	41.32	0.95

of Lena and its watermarked version are shown in Fig. 11 where difference between the two images is hardly visible. In Table 1, the PSNR and SSIM between the original images and their watermarked versions using the proposed algorithm and the algorithm proposed by Lee and Lin [12] are presented.

4.2 Payload

The payload represents the size of the watermark that can be hidden in the image in terms of the number of bits per pixel (bpp). In our proposed algorithm, the size of the watermark is a function of the image size and block size. Here, the block size is of 2 × 2. For each block, a 12-bit watermark is embedded. For an image of size $H \times W$, the total size of the watermark embedded in the image is $\frac{H \times W}{2 \times 2} \times 12$ bits with a payload of $\frac{12}{2 \times 2} = 3$ bpp.

4.3 Performance Against Tampering

To evaluate the effectiveness of the proposed scheme against tampering, localize the tampered regions, and restore them back as close as possible to the original, the

watermarked images were made to go through different types of tampers, viz. (1) *Direct Cropping* which can be classified into two sub-categories: (a) *cropping as a whole* where a single chunk is cropped from the image and (b) *multiple cropping* that includes *spread distribute* cropping where the cropping is spread all over the image and *chunk distribute* cropping where small number of relatively large chunks are cropped from the image; (2) *Object Insertion* where external objects are inserted into the watermarked image, and the object may be of large size, medium size, or small size; and (3) *Object Manipulation* where specific objects in the watermarked image are removed, destroyed, or changed.

Results of direct cropping (a) *Cropping as a whole*: Fig. 12 represents original image Lena of size 512 × 512, its watermarked version, different percentages of cropping attacks from center, and recovered images with their PSNR and SSIM values. From the result, we can see that the image can be restored up to a relatively good quality for cropping up to 60 %.

(b) *Multiple cropping*: Performance of the proposed scheme is evaluated against four different types of spread distribute tampering and eight different chunk distribute tampering. A total of 50 % of Peppers image is cropped. The cropped images along with corresponding tamper-localized and recovered images are shown in Fig. 13. Figure 13a_0–d_0 represents spread distribute tampering, while chunk distribute tampering is represented in Fig. 13e_0–l_0 for grayscale image of Peppers. The corresponding recovered images are presented in Fig. 13a_1–l_1 along with their PSNR values. For brevity, the same test image Peppers, as in [12], is taken into consideration so that conclusions can be drawn that for different tamper distributions too, our proposed scheme outperforms the one in [12].

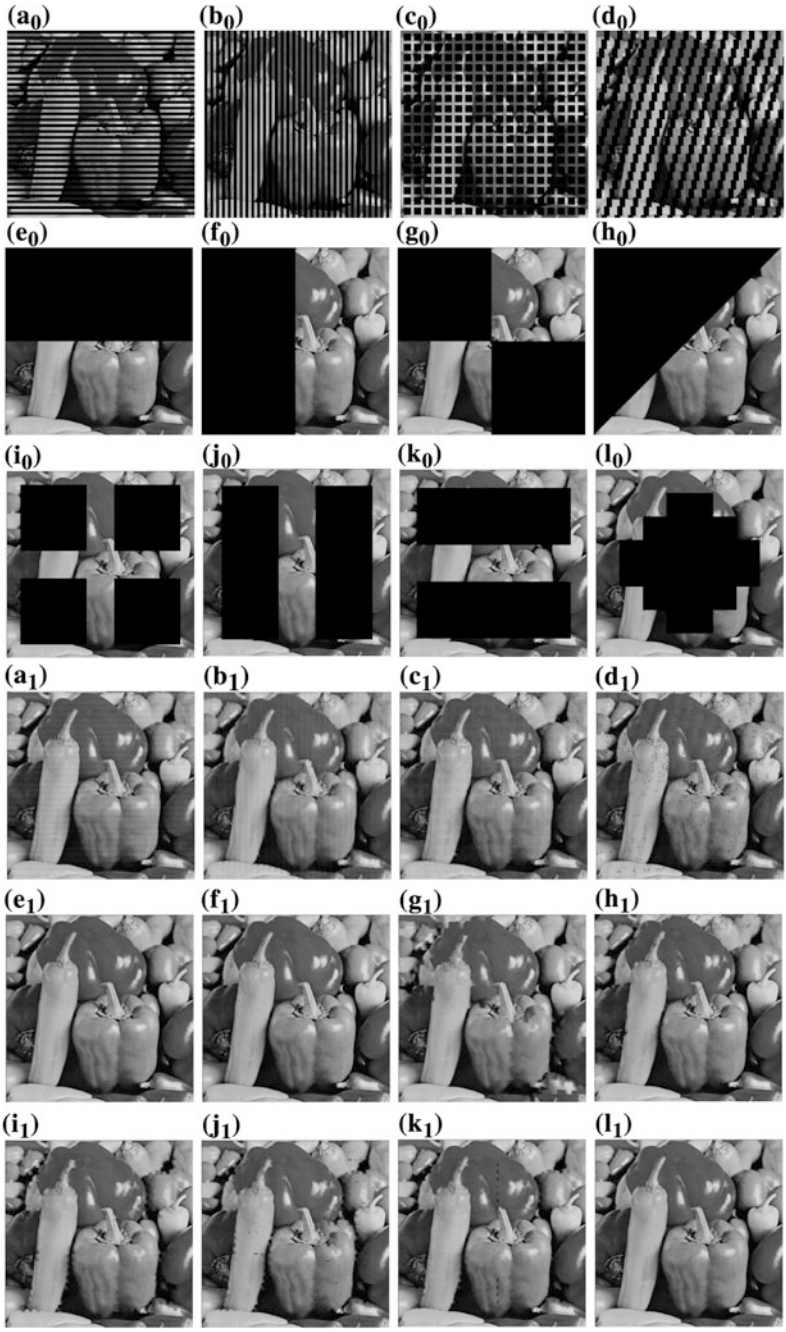
Results of object insertion One of the most common image tamperings by inserting objects is by copying/cutting regions of the watermarked image and pasting them into somewhere else in that image. The proposed watermarking system detects, localizes, and recovers the tampered regions of the images tampered by inserting small-, medium-, and large-sized objects as depicted in Fig. 14.

Results of object manipulation The watermarked image is attacked to remove, destroy, or change specific regions or objects in it. Figure 15 demonstrates three such attacks. The watermarked images are shown in Fig. 15a–c, the tampered images are shown in Fig. 15a_0–c_0, the tamper-localized images are shown in Fig. 15a_1–c_1, and the corresponding recovered images are shown in Fig. 15a_2–c_2.

4.4 Comparative Study

To examine the advantages of the proposed scheme over the existing techniques, a comparative study is presented in this section. As we employed a block-based spatial domain watermarking scheme, a well-known work in this field proposed by Lee and Lin [12] is taken into considerations for performance comparison. In our approach, we have used the three LSBs of each pixel in the image for watermark embedding where the watermark has been generated from the LL_1 sub-band of

Fig. 12 **a** Original Lena image; **b** watermarked image of (**a**) with PSNR = 41.44 and SSIM = 0.93; **c** image in (**b**) tampered by 25 % cropping at center; **d** recovered image from (**c**) with PSNR = 35.51 and SSIM = 0.90; **e** image in (**b**) tampered by 50 % cropping at center; **f** recovered image from (**e**) with PSNR = 30.91 and SSIM = 0.85; **g** image in (**b**) tampered by 60 % cropping at center; **h** recovered image from (**g**) with PSNR = 30.07 and SSIM = 0.82; **i** image in (**b**) tampered by 75 % cropping at center; **j** recovered image from (**i**) with PSNR = 27.55 and SSIM = 0.7645; **k** image in (**b**) tampered by 90 % cropping at center and **l** recovered image from (**k**) with PSNR = 24.91 and SSIM = 0.67


(a0)
(b0)
(c0)
(d0)
(e0)
(f0)
(g0)
(h0)
(i0)
(j0)
(k0)
(l0)
(a1)
(b1)
(c1)
(d1)
(e1)
(f1)
(g1)
(h1)
(i1)
(j1)
(k1)
(l1)

◀**Fig. 13** $\mathbf{a_0}$–$\mathbf{d_0}$ Spread distribute tampering, $\mathbf{e_0}$–$\mathbf{l_0}$ chunk distribute tampering of a total of 50 % in the watermarked image of Peppers (*grayscale*) of size 512 × 512, $\mathbf{a_1}$ recovered image of ($\mathbf{a_0}$) with PSNR = 32.19 dB, $\mathbf{b_1}$ recovered image of ($\mathbf{b_0}$) with PSNR = 30.58 dB, $\mathbf{c_1}$ recovered image of ($\mathbf{c_0}$) with PSNR = 33.12 dB, $\mathbf{d_1}$ recovered image of ($\mathbf{d_0}$) with PSNR = 28.76 dB, $\mathbf{e_1}$ recovered image of ($\mathbf{e_0}$) with PSNR = 32.89 dB, $\mathbf{f_1}$ recovered image of ($\mathbf{f_0}$) with PSNR = 33.30 dB, $\mathbf{g_1}$ recovered image of ($\mathbf{g_0}$) with PSNR = 27.56 dB, $\mathbf{h_1}$ recovered image of ($\mathbf{h_0}$) with PSNR = 30.19 dB, $\mathbf{i_1}$ recovered image of ($\mathbf{i_0}$) with PSNR = 29.30 dB, $\mathbf{j_1}$ recovered image of ($\mathbf{j_0}$) with PSNR = 29.95 dB, $\mathbf{k_1}$ recovered image of ($\mathbf{k_0}$) with PSNR = 31.39 dB, and $\mathbf{l_1}$ recovered image of ($\mathbf{l_0}$) with PSNR = 35.30 dB

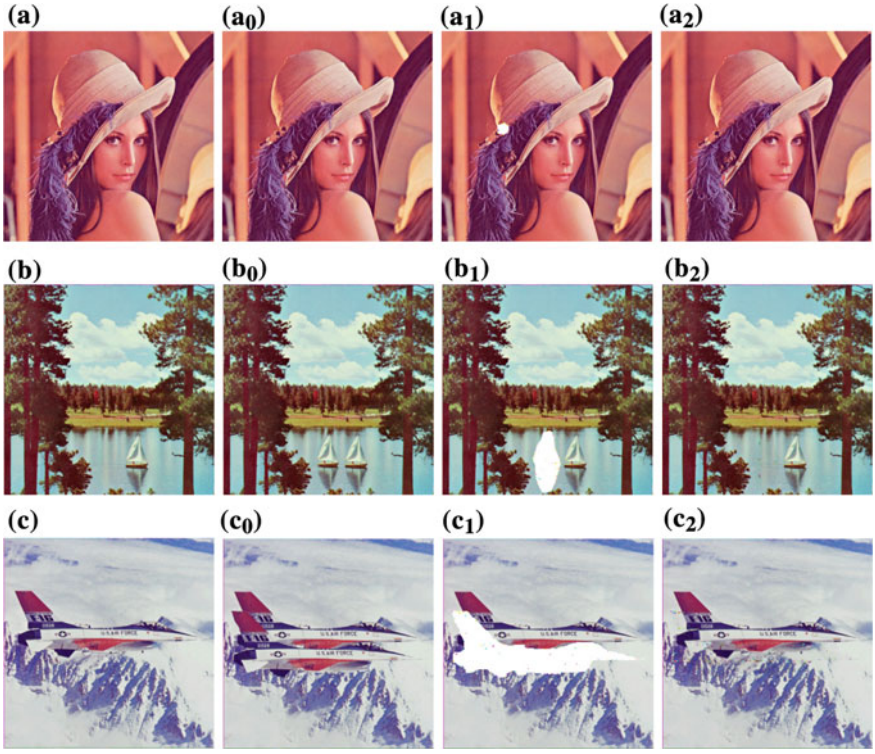


Fig. 14 Results of small-sized object insertion: **a** Watermarked image (*color*) of Lena of size 512 × 512, $\mathbf{a_0}$ tampered image of (**a**) by inserting small flower on the hat, $\mathbf{a_1}$ image in ($\mathbf{a_0}$) with localized tampered region, and $\mathbf{a_2}$ recovered image of ($\mathbf{a_0}$) with PSNR = 41.07 dB and SSIM index = 0.94. Results of medium-sized object insertion: **b** Watermarked image (*color*) of sailboat on lake of size 512 × 512, $\mathbf{b_0}$ tampered image of (**b**) by inserting a second sailboat on the lake, $\mathbf{b_1}$ image in ($\mathbf{b_0}$) with localized tampered region, and $\mathbf{b_2}$ recovered image of ($\mathbf{b_0}$) with PSNR = 39.61 dB and SSIM index = 0.0.95. Results of large-sized object insertion: **c** Watermarked image (*color*) of airplane of size 512 × 512, $\mathbf{c_0}$ tampered image of (**c**) by inserting a second F-16 airplane, $\mathbf{c_1}$ image in ($\mathbf{c_0}$) with localized tampered region, and $\mathbf{c_2}$ recovered image of ($\mathbf{c_0}$) with PSNR = 33.92 dB and SSIM index = 0.90

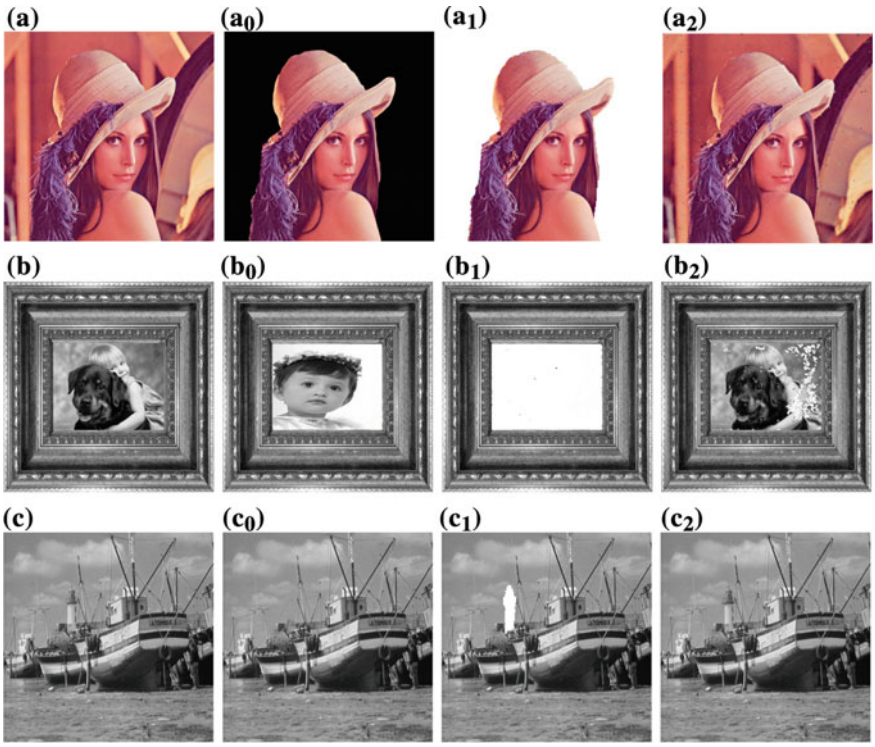


Fig. 15 **a** Watermarked image of Lena (*color*) of size 512 × 512, $\mathbf{a_0}$ tampered image of (**b**), $\mathbf{b_1}$ image of ($\mathbf{b_0}$) with localized tampered region, $\mathbf{a_2}$ recovered image of ($\mathbf{a_0}$) with PSNR = 32.90 dB and SSIM index = 0.87, **b** a sample watermarked image (*grayscale*) of size 512 × 512, $\mathbf{b_0}$ tampered image of (**b**), $\mathbf{b_1}$ image in ($\mathbf{b_0}$) with localized tampered region, $\mathbf{b_2}$ recovered image of ($\mathbf{b_0}$) with PSNR = 26.62 dB and SSIM index = 0.91, **c** watermarked image of boat (*grayscale*) of size 512 × 512, $\mathbf{c_0}$ tampered image of (**c**), $\mathbf{c_1}$ image of ($\mathbf{c_0}$) with localized tampered region, $\mathbf{c_2}$ recovered image of ($\mathbf{c_0}$) with PSNR = 40.53 dB and SSIM index = 0.95

DWT transformed blocks of the image. The quality of our watermarked image in terms of PSNR is around 41.2 dB, which is acceptable, and the distortion is imperceptible to HVS. In Table 1, the PSNR and SSIM between the original images and their watermarked versions using the proposed algorithm and the algorithm proposed by Lee and Lin [12] are presented. Table 2 lists the comparison of the PSNR of the recovered image for the sample grayscale image of Lena for various tampered sizes and locations. When the tampered region is as small as 2.34 %, the performance of [12] is better than ours. But when the amount of tampered region (in percentage) grows gradually, it can be inferred from Table 2 that the proposed method performs better than the one in [12]. Table 3 presents the comparative study of the average PSNR values of images recovered from cropping attacks of different sizes for all the images available in the misc volume of USC-SIPI [13] image database (color images are converted to their grayscale versions).

Table 2 PSNR of recovered image relative to the tampered size and location (*test image* Lena)

Tamper (crop %)	Tamper location	PSNR (in dB)	
		In Ref. [12]	Proposed
2.34	Top	48.09	41.37
2.4	Center	39.48	41.05
8.01	Corner	41.42	41.13
9.7	Center	35.17	40.08
25.0	Left	33.45	40.44
34.0	Top	33.01	40.06
40.1	Center	27.97	33.53
50.0	Center	26.59	30.91
65.0	Center	24.57	29.21
70.0	Center	24.16	28.28
75.0	Center	23.43	27.55
80.0	Center	22.55	25.83
85.0	Center	21.28	25.50
90.0	Center	19.86	24.91
95.0	Center	18.05	20.96
97.0	Center	16.87	19.65
Average PSNR		28.50	31.90

Table 3 Comparative analysis of PNSR of recovered images from cropping attacks of different sizes

Crop (%)	PSNR (in dB)	
	Proposed method	Ref. [12]
10	37.57	33.80
20	35.74	31.84
30	34.72	29.91
40	33.54	29.40
50	31.40	27.05
60	29.72	25.86
70	28.15	25.10
80	26.91	22.35
90	23.76	19.47
Average	31.28	27.20

5 Conclusion

The simulation of various kinds of tampering with different images has demonstrated the superiority of the proposed method over that of the existing ones for different extents of tampering. The embedding of the DWT-based watermark in four regions of the image has been the major contribution of this work. Embedding in multiple regions has made the approach robust and helped it to perform well in even severe cases of tampering. Further research is being conducted to improve its performance for situations where very small areas are tampered.

References

1. Li, K.F., Chen, T.S., Wu, S.C.: Image tamper detection and recovery system based on discrete wavelet transformation. In: IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, 26–28 Aug 2001. doi:10.1109/PACRIM.2001.953548 (2001)
2. Gang-chui, S., Mi-mi, Z.: Novel fragile authentication watermark based on chaotic system. In: International Symposium on Industrial Electronics, 4–7 May 2004. doi:10.1109/ISIE.2004.1572034 (2004)
3. Chen, T.S., Chen, J., Chen, J.G.: Tamper detection and retrieval technique based on JPEG2000 with LL subband. In Proceedings of IEEE International Conference OD Networking, Sensing & Control, Taipei, Taiwan (2004)
4. Tsai, P., Hu, Y.C.: A watermarking-based authentication with malicious detection and recovery. In: 5th International Conference on Information, Communications and Signal Processing. doi:10.1109/ICICS.2005.1689172 (2005)
5. Tsai, M.J., Chien, C.C.: A wavelet-based semi-fragile watermarking with recovery mechanism. In: IEEE International Symposium on Circuits and Systems, ISCAS 2008. doi:10.1109/ISCAS.2008.4542097 (2008)
6. Qi, X., Xin, X., Chang, R.: Image authentication and tamper detection using two complementary watermarks. In: 16th IEEE International Conference on Image Processing (ICIP). doi:10.1109/ICIP.2009.5413681 (2009)
7. Cruz, C., Mendoza, J.A., Miyatake, M.N., Meana, H.P., Kurkoski, B.: Semi-fragile watermarking based image authentication with recovery capability. In: International Conference on Information Engineering and Computer Science. doi:10.1109/ICIECS.2009.5363496 (2009)
8. Wang, N., Kim, C.W.: Tamper detection and self-recovery algorithm of color image based on robust embedding of dual visual watermarks using DWT-SVD. In: 9th International Symposium on Communications and Information Technology. doi:10.1109/ISCIT.2009.5341268 (2009)
9. Yuping, H., Guangjun, G.: Watermarking-based authentication with recovery mechanism. In: 2nd International Workshop on Computer Science and Engineering. doi:10.1109/WCSE.2009.856 (2009)
10. Hui, L., Yuping, H.: A wavelet-based watermarking scheme with authentication and recovery mechanism. In: International Conference on Electrical and Control Engineering (ICECE). doi:10.1109/iCECE.2010.86 (2010)
11. Wang, L.J., Syue, M.Y.: Image authentication and recovery using wavelet-based multipurpose watermarking. In: 10th International Joint Conference on Computer Science and Software Engineering (JCSSE). doi:10.1109/JCSSE.2013.6567315 (2013)

12. Lee, T., Lin, S.D.: Dual watermark for image tamper detection and recovery. Pattern Recogn. **41**, 3497–3506 (2008)
13. USC-SIPI image database: Available at http://sipi.usc.edu/database. Accessed on 1 Jan 2012
14. Computational Perception and Image Quality Lab, Oklahoma State University, www.vision.okstate.edu. Accessed on 1 Jan 2012
15. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)

Springer

http://www.springer.com/978-81-322-1987-3

Applied Computation and Security Systems
Volume Two
Chaki, R.; Saeed, K.; Choudhury, S.; Chaki, N. (Eds.)
2015, X, 211 p. 93 illus., Softcover
ISBN: 978-81-322-1987-3