Multimedia Tools and Applications
https://doi.org/10.1007/s11042-018-7084-0

A novel hash function based fragile watermarking method for image integrity

Ertugrul Gul[1,2] · Serkan Ozturk[2]

Received: 30 April 2018 / Revised: 30 October 2018 / Accepted: 18 December 2018
Published online: 10 January 2019
© Springer Science+Business Media, LLC, part of Springer Nature 2019

Abstract

In recent years, tampering and altering of digital images have become easier with the rapid development of computer technologies such as digital image editing tools. Therefore, verification of image integrity and tamper detection of digital images have become a great challenge. Fragile watermarking is the most widely used method for protecting the integrity and content authenticity of the image. In this paper, by using SHA-256 hash function, a novel block based fragile watermark embedding and tamper detection method is proposed. In watermark embedding phase, host image is divided into 32×32 non-overlapped blocks. Each 32×32 block is then divided into four 16×16 nonoverlapped sub-blocks. The entire hash value of the first three sub-blocks is generated as a watermark using SHA-256 hash function. The generated 256-bit binary watermark is embedded into the least significant bits (LSBs) of the fourth sub-block and watermarked image is obtained. In tamper detection phase, the detection of tampered block has been performed by comparing the hash value obtained from the three sub-blocks with the extracted watermark from the fourth sub-block of the watermarked image. The performance of the proposed method has been evaluated by applying linear and nonlinear attacks to the different regions of the watermarked images. Experimental results show that the proposed method detects all the tampered regions of the attacked images and high visual quality of watermarked images has been obtained.

Keywords Fragile image watermarking · SHA-256 · Image integrity

✉ Ertugrul Gul
ertugrulgul@erciyes.edu.tr

Serkan Ozturk
serkan@erciyes.edu.tr

[1] Computer Engineering Department, Nigde Omer Halisdemir University, 51240 Nigde, Turkey

[2] Computer Engineering Department, Erciyes University, Talas Cad, 38039 Kayseri, Turkey

1 Introduction

Due to the advances in computer networks and internet in recent years, the distribution and use of digital multimedia such as image and video has increased dramatically [7]. These multimedia data can easily be modified and tampered by using digital image editing tools [15]. Therefore, content authentication and verification of image integrity have become important. Signature based techniques and watermarking based techniques are widely used methods for protecting the integrity and content authenticity of the image [47]. In signature-based techniques, signature information of the image is obtained by using cryptography algorithms such as hash functions and this information is stored either in user-defined regions, or in a separate file [26, 35, 51]. In these techniques, the integrity verification can be performed by comparing the stored signature with the generated signature from the image. Hence, by using the comparison result, it can be determined whether the image is altered or not. However, drawback of the digital signature based techniques is that they can not detect and localize the tampered regions of the altered images [39, 47, 59]. On the other hand, in watermarking based techniques, binary logo, binary sequence or information data are embedded as a watermark on the unknown regions of the image. For the purpose of tamper detection, extracted watermarks are used to detect the destroyed or altered regions of the image. In addition to tamper detection, watermarking is also used in application areas such as ownership and copyright protection, content identification and management, digital forensic, fingerprinting, file archiving and broadcast monitoring [17, 45].

Digital watermarking techniques can be classified as robust, fragile and semi-fragile [20]. Robust watermarking techniques are mainly used for ownership identification, copyright protection and fingerprinting applications. Robust watermarks are designed to resist against intentional or unintentional attacks [1, 5, 54]. On the other hand, fragile watermarking techniques are used for the purpose of image forgery detection, content authentication and tamper detection. Therefore, fragile watermarks are designed to be easily destroyed or corrupted if a slight modification occurs on the watermarked image [7, 12, 30, 37–39, 58]. Moreover, semi-fragile watermarking techniques are used for content authentication and tamper detection applications. In semi-fragile image watermarking techniques, watermarks are designed to be robust for unintentional manipulations such as JPEG compression and are designed to be fragile against intentional attacks [2, 22, 34].

Fragile watermarking techniques can be categorized into two main classes namely spatial domain and transform domain. In spatial domain watermarking techniques, watermarks can be directly embedded by modifying the pixel values of the host image [8, 11, 33]. Least significant bit (LSB) substitution is one of the easiest and most popular method of the spatial domain techniques. In the transform domain watermarking techniques, watermark embedding is performed by modifying frequency coefficients of the host images. The most frequently used transform domain techniques in fragile watermarking are discrete cosine transform (DCT) [4, 6, 7, 24] and discrete wavelet transform (DWT) [23, 27].

With respect to the requirements of the application, fragile watermarking techniques should have major characteristics such as imperceptibility, fidelity, data payload, and blind detection [14, 43]. The imperceptibility means that the watermark should be invisible in the image and should not be perceived by the human visual system. The fidelity refers to watermark embedding process should not degrade the visual quality of the original host image. Data payload describes the number of bits that can be embedded into the host image [28]. Blind detection means that watermark extraction and tamper detection can be performed without any reference to the original host image [62].

Most of the fragile watermarking techniques use watermarks which are independent of the image content such as binary logo, binary sequence or identification data. In recent years, for the areas of medical and satellite imagery, hash based fragile watermarking methods have been improved. In hash based watermarking techniques, watermarks are generated by using hash value obtained from the image content. In the watermarking techniques developed for medical applications, information generated from the region of interest (ROI) of medical image is generally embedded into the region of noninterest (RONI) of the image. However, in these techniques, tamper detection can be performed inside ROI area. On the other hand, in applications where the entire image is important such as satellite images, the hash value obtained from all regions of the image is embedded in certain parts of the image. In most of these techniques, the entire hash value is not used due to the small size of the embedding area. For that reason, a certain amount of bits are removed from the hash value in order to perform embedding process. This creates vulnerability in protecting the image integrity. Moreover, in many watermarking techniques, the small size blocks are used for watermark embedding process. However, embedding large amount of data into small blocks reduces the visual quality of the watermarked images.

In this study, a hash based fragile watermarking method is proposed to overcome the above deficiencies. The whole host image is divided into 32×32 non-overlapped blocks in the proposed method. Then, each block is divided into four 16×16 sub-blocks. By using the SHA-256 hash function, the hash value of the first three sub-blocks is generated as a block watermark. This watermark is then embedded into the fourth sub-block by using LSB modification. Performance evaluation of the proposed method has been conducted by applying various attacks to the different regions of the watermarked images. The tampered blocks of the watermarked image have been detected by comparing the hash value obtained from the three sub-blocks with the extracted watermark from the fourth sub-block. The proposed method provides high imperceptibility and fidelity due to embedding watermark into quarter part of each 32×32 block with LSB modification. Moreover, the proposed method satisfies data payload requirement by using the 16×16 sub-block embedding area for 256 bit SHA hash value. Therefore, the use of entire hash value increases the reliability of the proposed method. Furthermore, tamper detection is performed without host image.

The main contributions of this paper are as followings: 1) we introduce a new fragile watermarking method based on SHA-256 hash function for image integrity and tamper detection; 2) we use entire hash value generated from three 16×16 sub-block of 32×32 non-overlapped image blocks as a watermark in order to improve the reliability of this method; 3) we achieve high visual quality of watermarked image (average 57 dB peak-signal-to-noise-ratio); 4) we apply various attacks to the watermarked images, and we clearly detect all the tampered regions.

The organization of the paper is as follows: Related works are introduced in Section 2. Section 3 describes the proposed method including the watermark embedding and the tamper detection. Experimental results are demonstrated in Section 4. The conclusion and the future work are given in Section 5.

2 Related work

In recent years, researchers have studied to improve the fragile image watermarking methods. Zhang and Wang [60] presented a hierarchical mechanism based fragile watermarking method, where block-derived and pixel-derived watermark data were embedded into the LSBs of the host image. Trivedy and Pal [50] proposed a fragile image watermarking method, in which watermark information and a key matrix were produced using a logistic map-based chaotic sequence. In this method, the watermark embedding was performed by using the key matrix. Overlapping block based and pixel based fragile image watermarking methods for tamper detection and content recovery was proposed by Qin et al. [40]. Authentication bits and reference bits were embedded into the image by LSB modification in this method. Singh and Singh [43] presented DCT based self embedded image watermarking method, where watermark was obtained from the five most significant bits of the pixels. Nazari et al. [29] proposed a fragile image watermarking method using chaotic maps, in which watermark was produced based on block characteristics of an image. Aslantas et al. [7] improved DCT based fragile image watermarking method, where rounding problems were corrected using intelligent optimization algorithms. A fast fractal compression coding and DCT based self-embedding fragile watermarking scheme for image authentication and recovery was proposed by Zhang et al. [61]. In this scheme, three types of watermarks were generated by using overlapped and interleaved image block structure. Singh and Agarwal [44] proposed a self recoverable dual watermarking method, where fragile and robust watermarks were embedded in different regions of the cover image for copyright protection and integrity verification. Li et al. [22] presented a wavelet group quantization and double authentication ring structure based semi-fragile self-recoverable watermarking scheme. The embedding ring was generated by a chaotic map and watermark was embedded into the mid-frequency bands by using significant difference parity quantization method, in this scheme.

Hash based fragile watermarking methods have also been developed by the researchers for image integrity verification and tamper detection. Hsu and Thu [14] proposed a probability based fragile watermarking method, in which authentication message obtained using MD5 hash function were embedded into the image by LSB substitution. Li et al. [21] presented a multi-block dependency based watermarking method for fingerprint images, where watermark was generated using MD5 hash function and encrypted by logistic chaotic map. Hong et al. [13] proposed a reversible image authentication method, in which authentication codes and MD5 hash values obtained from block features were embedded into the host image using pixel value ordering and LSB modification. Vasu et al. [53] presented DCT based image watermarking method, where hash values obtained by using Fast Johnson Lindenstrauss transform were embedded into the low frequency DCT coefficients. Das and Kundu [10] proposed a fragile image watermarking method for medical images. In this method, SHA-256 hash value of the ROI was used to verify the integrity of the image. Ustubioglu and Ulutas [52] presented medical image watermarking method, in which SHA-256 hash value of the center image area were embedded into the border image area using modified difference expansion and LSB techniques. Kunhu and Al-Ahmad [18] proposed a multiple watermarking method for color satellite images. Firstly, color watermark was embedded into the original host image in the frequency domain using DCT in this method. Then, SHA-256 based authentication hash watermark was embedded into the DCT watermarked image in the spatial domain using LSB modification. Khor et al. [16] presented a ROI based image watermarking method for ultrasound medical images. In this method, SHA-256 based hash value and recovery bits obtained from the ROI region were embedded into last two LSBs of the RONI of the image. Singh et al. [42] and Pandey et al. [32] proposed multiple text and image watermarking method for teleophthalmology using DWT and Singular Value Decomposition (SVD). In these methods, SHA-512 hash value of the ROI area was embedded into the singular values of

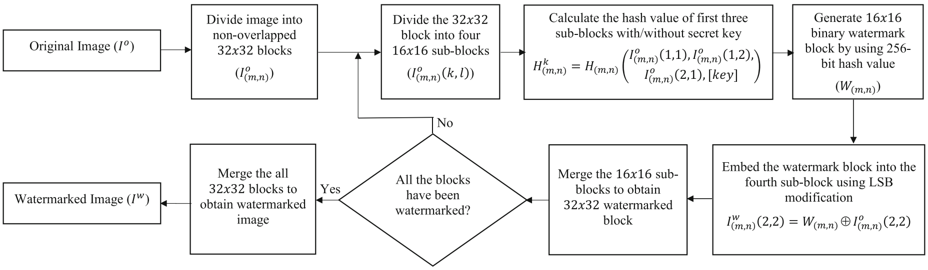


Fig. 1 Flowchart of the proposed watermark embedding method

the RONI part of the DWT images. Kunhu et al. [19] presented a reversible medical image watermarking method. The watermark obtained from the patients unique information and the hash value generated by SHA-256 hash function was embedded into the prediction error pixels by using the additive predictor error technique in this method. Loukhaoukha et al. [25] proposed a SVD and lifting wavelet transform based watermarking method, in which singular values of the binary watermark were embedded in a detail subband of host image. In this method, multiple scaling factors were optimized by using multi-objective ant colony optimization algorithm. Wahit et al. [55] presented an authentication technique, where MD5 signature was embedded in selected pixels of the host image by using LSB modification. A novel color image watermarking method based on Contourlet transform was proposed by Su et al. [49]. In this method, watermark embedding was performed by using MD5 hash function and Hessenberg decomposition. Aparna and Kishore [3] presented a medical image watermarking method based on SHA-256 and elliptical curve cryptography. Firstly, watermark was generated with the hash value of tumor part of medical image and encrypted electronic health record in this method. Then watermark was compressed and embedded by using Hybridization of

83	79	80	83	88	90	96	100	100	102	107	113	112	115	106	100	110	116	118	119	120	122	113	112	107	105	105	102	100	87	87	75
86	82	81	86	85	89	93	100	107	108	113	114	114	105	104	105	111	116	121	118	125	116	110	108	107	105	102	99	96	88	75	77
83	86	88	86	88	92	98	105	105	108	113	116	109	105	99	97	113	117	121	120	120	116	108	107	111	109	98	96	87	82	76	78
81	86	83	85	92	102	99	108	112	114	115	111	104	102	98	96	114	122	115	118	118	112	107	104	104	101	96	92	88	79	71	75
81	83	85	93	100	110	106	111	115	112	111	107	104	103	96	93	117	157	115	117	116	113	106	100	99	89	94	90	87	76	76	69
82	89	94	94	100	105	108	111	116	113	110	102	105	100	97	91	121	122	117	114	111	110	109	101	99	94	91	86	80	72	73	69
83	91	96	96	97	109	112	115	114	113	108	102	105	105	96	90	120	117	118	111	108	109	106	97	94	93	83	84	76	78	65	71
88	94	95	95	106	110	113	113	114	110	107	102	105	103	104	97	122	117	116	109	107	110	102	92	94	89	78	82	67	70	66	64
90	95	96	97	108	113	113	112	107	112	110	104	102	100	98	95	123	117	115	103	107	109	97	96	88	86	80	76	66	69	60	64
95	99	98	102	109	112	112	112	111	109	106	105	104	100	99	98	117	117	106	108	105	101	91	91	86	86	70	67	65	61	59	56
99	102	106	106	110	119	112	111	111	111	106	105	105	100	100	98	116	111	110	106	99	96	92	84	79	75	71	67	72	63	54	53
102	108	106	109	117	117	113	115	118	112	105	103	104	102	94	90	117	109	109	105	97	88	86	86	72	65	63	57	61	64	56	61
105	114	107	112	114	115	119	117	112	114	105	102	100	106	94	88	120	111	103	98	90	85	83	77	67	65	63	57	63	54	60	58
103	110	112	113	118	114	114	113	113	110	101	102	104	98	93	87	116	106	98	91	83	83	76	70	65	62	58	61	55	55	57	55
106	111	109	111	113	118	121	115	117	113	107	109	106	95	92	83	113	103	98	88	80	69	67	66	66	65	59	59	53	51	52	58
114	111	107	114	116	120	115	111	110	105	109	102	98	94	88	83	113	103	98	88	80	69	67	66	66	65	59	59	53	51	52	58
98	96	88	87	87	90	90	86	81	69	68	66	64	66	74	72	80	76	72	67	64	64	68	70	72	68	65	63	64	63	66	57
102	94	89	88	85	86	87	78	76	66	66	69	70	69	73	70	73	69	68	69	68	67	62	70	66	62	65	59	60	65	60	60
95	97	89	86	86	81	84	74	75	68	66	67	77	78	77	76	67	66	65	67	63	71	64	62	59	61	58	64	60	62	55	54
96	100	90	92	85	78	74	74	70	61	70	69	74	79	76	75	64	67	62	67	67	62	65	62	58	61	62	57	54	57	56	63
96	96	93	86	83	80	81	73	72	69	72	73	78	72	67	66	70	68	62	66	66	65	72	64	62	64	64	64	59	63	61	60
100	93	92	85	84	79	74	66	69	74	77	75	73	69	63	67	68	62	64	65	60	66	69	58	60	63	60	63	59	64	72	71
95	98	91	79	88	77	65	66	71	68	77	83	81	75	71	64	61	66	64	70	66	65	63	60	62	68	64	65	63	66	76	86
95	96	84	78	78	72	62	71	74	70	75	78	75	80	70	65	65	57	60	68	73	69	61	62	58	55	58	70	67	74	80	95
95	87	81	78	70	64	65	69	80	75	75	76	77	75	66	62	60	58	61	63	65	65	66	80	64	60	68	77	76	77	82	90
91	83	75	73	64	71	69	79	75	74	74	80	76	71	69	65	54	56	59	59	61	64	60	75	66	68	77	82	86	86	93	96
90	84	72	73	67	68	74	82	79	73	71	72	71	67	67	63	56	66	63	63	63	65	73	72	65	73	80	82	99	93	91	97
84	75	72	71	71	74	67	74	75	68	73	75	68	59	64	56	62	58	61	61	66	60	63	70	73	83	87	92	93	96	96	92
78	78	74	72	73	69	67	70	71	73	71	73	65	65	58	57	61	58	61	62	64	68	69	76	80	85	92	95	101	103	99	98
79	79	77	66	73	72	69	63	66	67	70	63	63	64	57	54	57	60	55	60	61	66	76	75	84	85	95	99	101	105	101	100
78	75	75	73	69	69	70	65	68	72	68	60	62	61	59	56	55	60	62	66	74	78	81	87	97	94	99	106	102	104	106	108
81	80	71	70	68	67	66	67	66	68	71	62	62	60	70	63	55	60	62	66	74	78	81	87	97	94	99	106	102	104	106	108

Fig. 2 Original 32 × 32 sample block

$H^k_{(m,n)}$	6777535be0684fc12507c70c55db0c728f776aee2ebe57a82874dfec955206a5

Fig. 3 Calculated hash value of the first three 16 × 16 sub-blocks with secret key

Compression algorithm. Su and Chen [48] proposed a blind color image watermarking method, in which the binary watermark was embedded into the blue layer of a RGB host image in the spatial domain by using MD5 hash function.

Most of the hash based fragile watermarking techniques are used in the area of medical imagery where the patient data is important. Hash based medical image watermarking techniques split the medical image into two regions such as ROI and RONI. The hash value obtained from the ROI area of the medical image is generally embedded as a watermark into the RONI area. However, these techniques only verify the integrity of ROI and detect the tampered blocks inside ROI. For that reason, these type of watermarking techniques are inappropriate for military and remote sensing areas where the entire image is important. On the other hand, in satellite image watermarking techniques, the hash value obtained from all regions of the image are embedded in certain parts of the image. In most of these techniques, the entire hash value can not be used as a watermark due to the small size of the embedding area. Therefore, watermark or authentication information is obtained by removing a certain amount of bits from the hash value. This situation creates vulnerability in verification of image integrity. Moreover, most of the hash based watermarking techniques use small size blocks for watermark embedding. However, embedding large amount of authentication data into small blocks decreases the visual quality of the watermarked images. Therefore, in these studies, obtained visual quality (PSNR) value ranged from around 35 *dB* to around 50 *dB*.

In this paper an efficient *32* × *32* block based fragile watermarking method is proposed by using SHA-256 hash function. Unlike the aforementioned methods, quarter part of each *32* × *32* block is used for watermark embedding area in this method. In addition, generated 256 bit hash value is completely embedded into the *16* × *16* sub-block embedding area for each *32* × *32* block in order to increase the reliability of the proposed method. By using proposed block based scheme and LSB modification, better visual quality of watermarked images (PSNR values

0	1	1	0	0	1	1	1	0	1	1	1	0	1	1	1
0	1	0	1	0	0	1	1	0	1	0	1	1	0	1	1
1	1	1	0	0	0	0	0	0	1	1	0	1	0	0	0
0	1	0	0	1	1	1	1	1	1	0	0	0	0	0	1
0	0	1	0	0	1	0	1	0	0	0	0	0	1	1	1
1	1	0	0	0	1	1	1	0	0	0	0	1	1	0	0
0	1	0	1	0	1	0	1	1	1	0	1	1	0	1	1
0	0	0	0	1	1	0	0	0	1	1	1	0	0	1	0
1	0	0	0	1	1	1	1	0	1	1	1	0	1	1	1
0	1	1	0	1	0	1	0	1	1	1	0	1	1	1	0
0	0	1	0	1	1	1	0	1	0	1	1	1	1	1	0
0	1	0	1	0	1	1	1	1	0	1	0	1	0	0	0
0	0	1	0	1	0	0	0	0	1	1	1	0	1	0	0
1	1	0	1	1	1	1	1	1	1	1	0	1	1	0	0
1	0	0	1	0	1	0	1	0	1	0	1	0	0	1	0
0	0	0	0	0	1	1	0	1	0	1	0	0	1	0	1

Fig. 4 Generated watermark block

83	79	80	83	88	90	96	100	100	102	107	113	112	115	106	100	110	116	118	119	120	122	113	112	107	105	105	102	100	87	87	75
86	82	81	86	85	89	93	100	107	108	113	114	114	105	104	105	111	116	121	118	125	116	110	108	107	105	102	99	96	88	75	77
83	86	88	86	88	92	98	105	105	108	113	116	109	105	99	97	113	117	121	120	120	116	108	107	111	109	98	96	87	82	76	78
81	86	83	85	92	102	99	108	112	114	115	111	104	102	98	96	114	122	115	118	118	112	107	104	104	101	96	92	88	79	71	75
81	83	85	93	100	110	106	111	115	112	111	107	104	103	96	93	117	157	115	117	116	113	106	100	99	89	94	90	87	76	76	69
82	89	94	94	100	105	108	111	116	113	110	102	105	100	97	91	121	122	117	114	111	110	109	101	99	94	91	86	80	72	73	69
83	91	96	96	97	109	112	115	114	113	108	102	105	105	96	90	120	117	118	111	108	109	106	97	94	93	83	84	76	78	65	71
88	94	95	95	106	110	113	113	114	110	107	102	105	103	104	97	122	117	116	109	107	110	102	92	94	89	78	82	67	70	66	64
90	95	96	97	108	113	113	112	107	112	110	104	102	100	98	95	123	117	115	103	107	109	97	96	88	86	80	76	66	69	60	64
95	99	98	102	109	112	112	112	111	109	106	105	104	100	99	98	117	117	106	108	105	101	91	91	86	86	70	67	65	61	59	56
99	102	106	106	110	119	112	111	111	111	106	105	105	100	100	98	116	111	110	106	99	96	92	84	79	75	71	67	72	63	54	53
102	108	106	109	117	117	113	115	118	112	105	103	104	102	94	90	117	109	109	105	97	88	86	86	72	65	63	57	61	64	56	61
105	114	107	112	114	115	119	117	112	114	105	102	100	106	94	88	120	111	103	98	90	85	83	77	67	65	63	57	63	54	60	58
103	110	112	113	118	114	114	113	113	110	101	102	104	98	93	87	116	106	98	91	83	83	76	70	65	62	58	61	55	55	57	55
106	111	109	111	113	118	121	115	117	113	107	109	106	95	92	83	113	103	98	88	80	69	67	66	66	65	59	59	53	51	52	58
114	111	107	114	116	120	115	111	110	105	109	102	98	94	88	83	113	103	98	88	80	69	67	66	66	65	59	59	53	51	52	58
98	96	88	87	87	90	90	86	81	69	68	66	64	66	74	72	80	77	73	66	64	65	69	71	72	69	65	63	64	63	67	57
102	94	89	88	85	86	87	78	76	66	66	69	70	69	73	70	72	69	68	69	68	66	63	71	66	63	64	59	61	64	61	61
95	97	89	86	86	81	84	74	75	68	66	67	77	78	77	76	67	67	65	66	62	70	64	62	58	61	59	64	61	62	54	54
96	100	90	92	85	78	74	74	70	61	70	69	74	79	76	75	64	67	62	66	67	63	65	63	59	61	62	56	54	56	56	63
96	96	93	86	83	80	81	73	72	69	72	73	78	72	67	66	70	68	63	66	66	65	72	65	62	64	64	64	58	63	61	61
100	93	92	85	84	79	74	66	69	74	77	75	73	69	63	67	69	63	64	64	60	67	69	59	60	62	60	62	59	65	72	70
95	98	91	79	88	77	65	66	71	68	77	83	81	75	71	64	60	67	64	71	66	65	62	61	63	69	64	65	63	66	77	87
95	96	84	78	78	72	62	71	74	70	75	78	75	80	70	65	64	56	60	68	73	69	60	62	58	55	59	71	66	74	81	94
95	87	81	78	70	64	65	69	80	75	75	76	77	75	66	62	61	58	60	62	65	65	67	81	64	61	69	77	76	77	83	91
91	83	75	73	64	71	69	79	75	74	74	80	76	71	69	65	54	57	59	58	61	64	61	74	67	69	77	82	87	87	93	96
90	84	72	73	67	68	74	82	79	73	71	72	71	67	67	63	56	66	63	62	63	65	73	72	65	72	81	83	99	93	91	96
84	75	72	71	71	74	67	74	75	68	73	75	68	59	64	56	62	59	60	61	66	61	63	71	73	82	87	92	93	96	96	92
78	78	74	72	73	69	67	70	71	73	71	73	65	65	58	57	60	58	61	62	65	68	68	76	80	85	93	95	100	103	98	98
79	79	77	66	73	72	69	63	66	67	70	63	63	64	57	54	57	61	54	61	61	67	77	75	85	85	95	98	101	105	100	100
78	75	75	73	69	69	70	65	68	72	68	60	62	61	59	56	55	60	62	67	74	79	80	87	96	95	98	107	102	104	107	108
81	80	71	70	68	67	66	67	66	68	71	62	62	60	70	63	54	60	62	66	74	79	81	86	97	94	99	106	102	105	106	109

Fig. 5 Watermarked 32 × 32 block

above 57 *dBs*) are obtained than the related works. Finally, experimental results show that the proposed method can successfully detect all the tampered regions of the attacked images.

3 Proposed method

Hash function based fragile image watermarking method is proposed to protect image integrity and to detect tampered regions. Irrespective of the data size, SHA-256 hash function generates 256-bit hash value [36]. The proposed block based fragile watermarking algorithm uses *16* × *16* blocks for watermark embedding. Therefore, SHA-256 hash function is used to generate 256-bit hash value as a watermark. The proposed method consists of two main parts; watermark embedding and tamper detection.

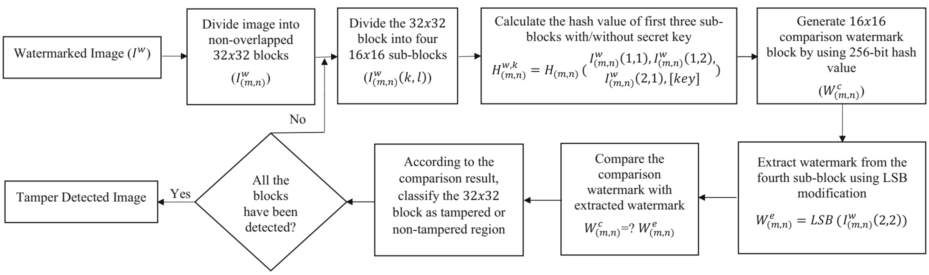


Fig. 6 Flow chart of the proposed tamper detection method

83	79	80	83	88	90	96	100	100	102	107	113	112	115	106	100	110	116	118	119	120	122	113	112	107	105	105	102	100	87	87	75
86	82	81	86	85	89	93	100	107	108	113	114	114	105	104	105	111	116	121	118	125	116	110	108	107	105	102	99	96	88	75	77
83	86	88	86	88	92	98	105	105	108	113	116	109	105	99	97	113	117	121	120	120	116	108	107	111	109	98	96	87	82	76	78
81	86	83	85	92	102	99	108	112	114	115	111	104	102	98	96	114	122	115	118	118	112	107	104	104	101	96	92	88	79	71	75
81	83	85	93	100	110	106	111	115	112	111	107	104	103	96	93	117	157	115	117	116	113	106	100	99	89	94	90	87	76	76	69
82	89	94	94	100	105	108	111	116	113	110	102	105	100	97	91	121	122	117	114	111	110	109	101	99	94	91	86	80	72	73	69
83	91	96	96	97	109	112	115	114	113	108	102	105	105	96	90	120	117	118	111	108	109	106	97	94	93	83	84	76	78	65	71
88	94	95	95	106	110	113	113	114	110	107	102	105	103	104	97	122	117	116	109	107	110	102	92	94	89	78	82	67	70	66	64
90	95	96	97	108	113	113	112	107	112	110	104	102	100	98	95	123	117	115	103	107	109	97	96	88	86	80	76	66	69	60	64
95	99	98	102	109	112	112	112	111	109	106	105	104	100	99	98	117	117	106	108	105	101	91	91	86	86	70	67	65	61	59	56
99	102	106	106	110	119	112	111	111	111	106	105	105	100	100	98	116	111	110	106	99	96	92	84	79	75	71	67	72	63	54	53
102	108	106	109	117	117	113	115	118	112	105	103	104	102	94	90	117	109	109	105	97	88	86	86	72	65	63	57	61	64	56	61
105	114	107	112	114	115	119	117	112	114	105	102	100	106	94	88	120	111	103	98	90	85	83	77	67	65	63	57	63	54	60	58
103	110	112	113	118	114	114	113	113	110	101	102	104	98	93	87	116	106	98	91	83	83	76	70	65	62	58	61	55	55	57	55
106	111	109	111	113	118	121	115	117	113	107	109	106	95	92	83	113	103	98	88	80	69	67	66	66	65	59	59	53	51	52	58
114	111	107	114	116	120	115	111	110	105	109	102	98	94	88	83	113	103	98	88	80	69	67	66	66	65	59	59	53	51	52	58
98	96	87	87	87	90	90	86	81	69	68	66	64	66	74	72	80	77	73	66	64	65	69	71	72	69	65	63	64	63	67	57
102	94	89	88	85	86	87	78	76	66	66	69	70	69	73	70	72	69	68	69	68	66	63	71	66	63	64	59	61	64	61	61
95	97	89	86	86	81	84	74	75	68	66	67	77	78	77	76	67	67	65	66	62	70	64	62	58	61	59	64	61	62	54	54
96	100	90	92	85	78	74	74	70	61	70	69	74	79	76	75	64	67	62	66	67	63	65	63	59	61	62	56	54	56	56	63
96	96	93	86	83	80	81	73	72	69	72	73	78	72	67	66	70	68	63	66	66	65	72	65	62	64	64	64	58	63	61	61
100	93	92	85	84	79	74	66	69	74	77	75	73	69	63	67	69	63	64	64	60	67	69	59	60	62	60	62	59	65	72	70
95	98	91	79	88	77	65	66	71	68	77	83	81	75	71	64	60	67	64	71	66	65	62	61	63	69	64	65	63	66	77	87
95	96	84	78	78	72	62	71	74	70	75	78	75	80	70	65	64	56	60	68	73	69	60	62	58	55	59	71	66	74	81	94
95	87	81	78	70	64	65	69	80	75	75	76	77	75	66	62	61	58	60	62	65	65	67	81	64	61	69	77	76	77	83	91
91	83	75	73	64	71	69	79	75	74	74	80	76	71	69	65	54	57	59	58	61	64	61	74	67	69	77	82	87	87	93	96
90	84	72	73	67	68	74	82	79	73	71	72	71	67	67	63	56	66	63	62	63	65	73	72	65	72	81	83	99	93	91	96
84	75	72	71	71	74	67	74	75	68	73	75	68	59	64	56	62	59	60	61	66	61	63	71	73	82	87	92	93	96	96	92
78	78	74	72	73	69	67	70	71	73	71	73	65	65	58	57	60	58	61	62	65	68	68	76	80	85	93	95	100	103	98	98
79	79	77	66	73	72	69	63	66	67	70	63	63	64	57	54	57	61	54	61	61	67	77	75	85	85	95	98	101	105	100	100
78	75	75	73	69	69	70	65	68	72	68	60	62	61	59	56	55	60	62	67	74	79	80	87	96	95	98	107	102	104	107	108
81	80	71	70	68	67	66	67	66	68	71	62	62	60	70	63	54	60	62	66	74	79	81	86	97	94	99	106	102	105	106	109

Fig. 7 Tampered 32 × 32 block

3.1 Watermark embedding method

The flowchart of the proposed watermark embedding method is given in Fig. 1. The gray-scale image of size *MxN* is first divided into non-overlapped blocks of size *32* × *32*. Then, these blocks are divided into four non-overlapped sub-blocks of size *16* × *16*. For public watermarking applications, the unique SHA-256 hash value of the first three sub-blocks is calculated as a block watermark for each *32* × *32* block. The generated 256-bit watermark is then embedded into the LSBs of the fourth sub-block of size *16* × *16*.On the other hand, optionally a secret key may also be used to improve privacy in private watermarking applications. The basic steps of the proposed method are as follows:

1. Divide original image of size *MxN* into *32* × *32* non-overlapped blocks:

$$I^o = \bigcup_{m=1}^{M/32} \bigcup_{n=1}^{N/32} I^o_{(m,n)}$$

2. Divide each image block of size *32* × *32* into *16* × *16* non-overlapped sub-blocks:

$$I^o_{(m,n)} = \bigcup_{k=1}^{32/16} \bigcup_{l=1}^{32/16} I^o_{(m,n)}(k,l)$$

$H^{w,k}_{(m,n)}$	2a6164373675aaf52e8c60846f9681873c2c7780e18707f6e58ed59ad7a26e7c

Fig. 8 Calculated hash value of the first three watermarked sub-blocks with secret key

0	0	1	0	1	0	1	0	0	1	1	0	0	0	0	1
0	1	1	0	0	1	0	0	0	0	1	1	0	1	1	1
0	0	1	1	0	1	1	0	0	1	1	1	0	1	0	1
1	0	1	0	1	0	1	0	1	1	1	1	0	1	0	1
0	0	1	0	1	1	1	0	1	0	0	0	1	1	0	0
0	1	1	0	0	0	0	0	1	0	0	0	0	1	0	0
0	1	1	0	1	1	1	1	1	0	0	1	0	1	1	0
1	0	0	0	0	0	0	1	1	0	0	0	0	1	1	1
0	0	1	1	1	1	0	0	0	0	1	0	1	1	0	0
0	1	1	1	0	1	1	1	1	0	0	0	0	0	0	0
1	1	1	0	0	0	0	1	1	0	0	0	0	1	1	1
0	0	0	0	0	1	1	1	1	1	1	1	0	1	1	0
1	1	1	0	0	1	0	1	1	0	0	0	1	1	1	0
1	1	0	1	0	1	0	1	1	0	0	1	1	0	1	0
1	1	0	1	0	1	1	1	1	0	1	0	0	0	1	0
0	1	1	0	1	1	1	0	0	1	1	1	1	1	0	0

Fig. 9 Generated comparison watermark block

3. Calculate the hash value of the first three sub-blocks with/without secret key by using SHA-256 hash function:

$$H^k_{(m,n)} = H_{(m,n)}\left(I^o_{(m,n)}(1,1), I^o_{(m,n)}(1,2), I^o_{(m,n)}(2,1), [key]\right)$$

4. Generate binary watermark of size 16×16 using 256-bit hash value:

$$W_{(m,n)} = H^k_{(m,n)}$$

5. Embed the watermark into the fourth sub-block using LSB modification:

$$I^w_{(m,n)}(2,2) = W_{(m,n)} \oplus I^o_{(m,n)}(2,2)$$

6. Repeat steps 2-5 for all 32×32 blocks.

0	1	1	0	0	1	1	1	0	1	1	1	0	1	1	1
0	1	0	1	0	0	1	1	0	1	0	1	1	0	1	1
1	1	1	0	0	0	0	0	0	1	1	0	1	0	0	0
0	1	0	0	1	1	1	1	1	1	0	0	0	0	0	1
0	0	1	0	0	1	0	1	0	0	0	0	0	1	1	1
1	1	0	0	0	1	1	1	0	0	0	0	1	1	0	0
0	1	0	1	0	1	0	1	1	1	0	1	1	0	1	1
0	0	0	0	1	1	0	0	0	1	1	1	0	0	1	0
1	0	0	0	1	1	1	1	0	1	1	1	0	1	1	1
0	1	1	0	1	0	1	0	1	1	1	0	1	1	1	0
0	0	1	0	1	1	1	0	1	0	1	1	1	1	1	0
0	1	0	1	0	1	1	1	1	0	1	0	1	0	0	0
0	0	1	0	1	0	0	0	0	1	1	1	0	1	0	0
1	1	0	1	1	1	1	1	1	1	1	0	1	1	0	0
1	0	0	1	0	1	0	1	0	1	0	1	0	0	1	0
0	0	0	0	0	1	1	0	1	0	1	0	0	1	0	1

Fig. 10 Extracted watermark

Fig. 11 Original images of size 512×512: a) Lena, b) Boat, c) Man, and d) Lake

Figures 2, 3, 4 and 5 illustrate an example of the watermark embedding process. In this example, '*GUL_OZTURK*' is used as a secret key. Figure 2 shows a sample block of size 32×32 that chosen randomly from the original image. By using the SHA-256 hash function, the calculated hash value of the first three 16×16 sub-blocks with the secret key is given in Fig. 3. Generated 16×16 binary watermark block is shown in Fig. 4. The watermarked 32×32 block obtained by embedding the watermark block into fourth sub-block using LSB modification is given in Fig. 5.

3.2 Tamper detection method

The flowchart of the proposed tamper detection method is illustrated in Fig. 6. The watermarked image is first divided into 32×32 non-overlapped blocks. These watermarked blocks are then divided into four 16×16 sub-blocks. The hash value of first three sub-blocks with/without a secret key is generated as a comparison watermark by using SHA-256 hash function. Then, watermark is extracted from the watermarked fourth sub-block using LSB modification. Finally, detection of the tampered blocks is performed by comparing the extracted watermark with the comparison watermark. The basic steps of the tamper detection method are as follows:

1. Divide the watermarked image of size *MxN* into 32×32 non-overlapped blocks:

$$I^w = \bigcup_{m=1}^{M/32} \bigcup_{n=1}^{N/32} I^w_{(m,n)}$$

2. Divide each watermarked block into 16×16 non-overlapped sub-blocks:

$$I^w_{(m,n)} = \bigcup_{k=1}^{32/16} \bigcup_{l=1}^{32/16} I^w_{(m,n)}(k,l)$$

3. Calculate the hash value of the first three watermarked sub-blocks with/without secret key by using SHA-256 hash function:

Fig. 12 Watermarked images: a) Lena, b) Boat, c) Man, and d) Lake

Table 1 The PSNR results

Image	Lena	Boat	Man	Lake
PSNR (*dB*)	57.196	57.161	57.151	57.150

$$H^{w,k}_{(m,n)} = H_{(m,n)}\left(I^{w}_{(m,n)}(1,1), I^{w}_{(m,n)}(1,2), I^{w}_{(m,n)}(2,1), [key]\right)$$

4. Generate comparison binary watermark of size 16×16 using 256-bit hash value:

$$W^{c}_{(m,n)} = H^{w,k}_{(m,n)}$$

5. Extract watermark from the fourth sub-block using LSB modification:

$$W^{e}_{(m,n)} = LSB\left(I^{w}_{(m,n)}(2,2)\right)$$

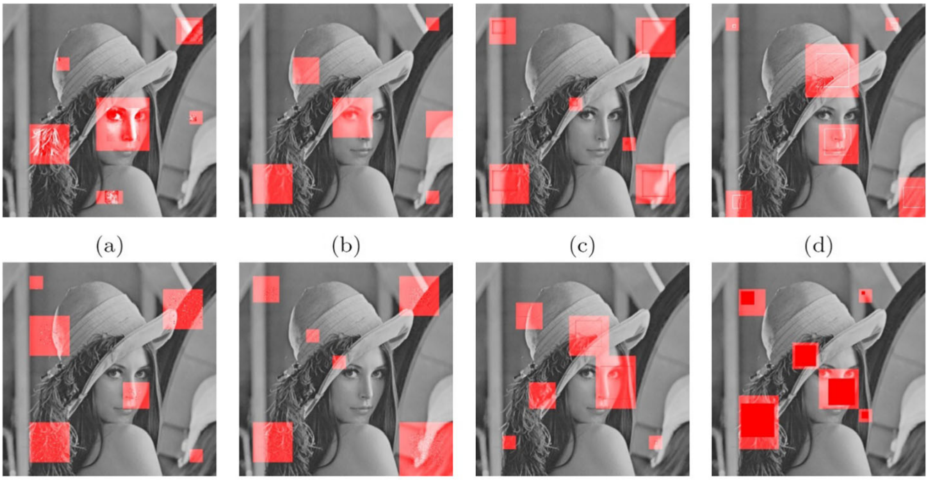
6. Compare the comparison watermark with extracted watermark:

$$W^{c}_{(m,n)} = ?W^{e}_{(m,n)}$$

7. Classify the 32×32 block as tampered or non-tampered region:
8. Repeat steps 2-7 for all 32×32 blocks.

An example of tamper detection process is given in Figs. 7, 8, 9 and 10. One-bit modification is applied to the third sub-block of watermarked image that is shown in Fig. 5. Tampered 32×32 image block is given in Fig. 7. Figure 8 shows the calculated hash value of the first three watermarked sub-blocks with the secret key. Generated comparison watermark

Fig. 13 Implementation of random sized attacks to the different regions of the watermarked Lena image: a) HE, b) scaling, c) blurring, d) sharpen, e) SPNA, f) GNA, g) AF, h) cropping



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 14 Detection of the tampered regions of the watermarked Lena image: a) HE, b) scaling, c) blurring, d) sharpen, e) SPNA, f) GNA, g) AF, h) cropping

block is given in Fig. 9. Extracted watermark from the LSBs of the watermarked fourth sub-block is shown in Fig. 10. Comparison results are also highlighted in this figure. The figure indicates that the proposed method detects one-bit modification.

The computational complexity of the watermark technique refers to the cost of embedding and extracting watermark [41]. In the proposed method, watermark embedding and tamper detection are performed in the spatial domain. The computational complexity of our method is low compared to transform domain watermarking methods [31, 46]. The required processing mainly lies on dividing blocks and sub-blocks, generating hash value, embedding or extracting LSBs of the sub-blocks, and comparing the bits. For *MxN* image, time cost of the proposed method is determined by dividing blocks. A hash value is generated for each of the 32×32 non-overlapped block, which takes *O(M/32xN/32xH)* time. H is the computational complexity of the SHA-256 hash function. The 256-bit hash value is embedded or extracted to/from 16×16 sub block, which takes $O(16 \times 16)$ time. Thus, the overall computational complexity of the algorithm is *O(M/2xN/2xH)*.

(a) (b) (c)

Fig. 15 Implementation of different attacks to the different 30×30 regions of the watermarked images: a) Boat, b) Man, c) Lake

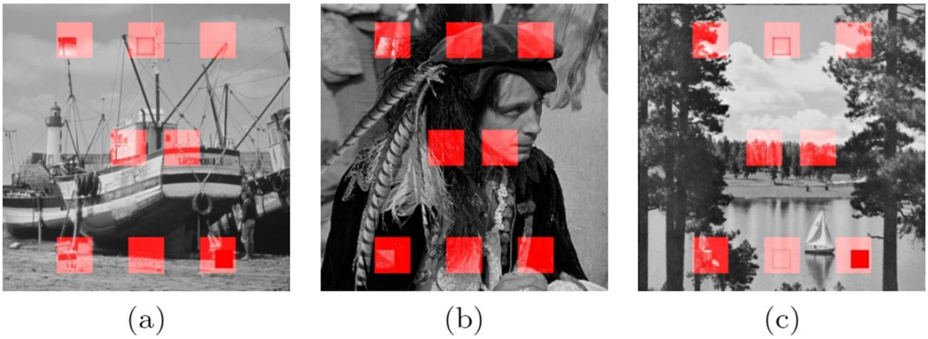


Fig. 16 Detection of the tampered regions of the watermarked images: a) Boat, b) Man, c) Lake

4 Experimental results

In order to demonstrate the performance of the proposed method, 512×512 Lena, Boat, Man and Lake images shown in Fig. 11 are used. The images that are watermarked by the proposed method are illustrated in Fig. 12. The secret key is not used in these samples.

To determine the quantitative degradation of the image, PSNR are calculated using the equation as given below:

$$PSNR = 10log_{10}\left(\frac{255^2}{\left(\frac{1}{mxn}\right)\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left(I^o_{i,j}-I^w_{i,j}\right)^2}\right)$$

where, mxn is the size of the original image, $I^o_{(i,j)}$ and $I^w_{(i,j)}$ are the pixel values at position (i,j) of the original image $I^o$and the watermarked image I^w, respectively.

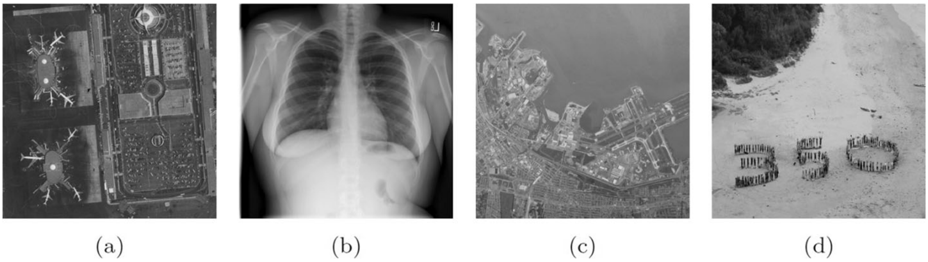


Fig. 17 Original images of 1024×1024: a) Airport, b) ChestX-ray, c) San Francisco, d) Threehundred

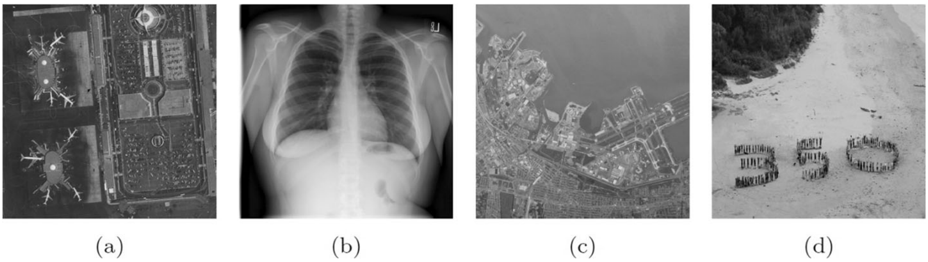


Fig. 18 Watermarked images of 1024×1024: a) Airport, b) ChestX-ray, c) San Francisco, d) Threehundred

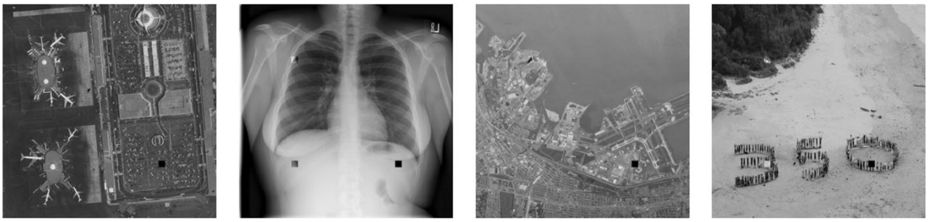
Table 2 The PSNR results of 1024×1024 images

Image	Airport	ChestX-ray	San Francisco	Threehundred
PSNR (*dB*)	57.156	57.165	57.172	57.161

The PSNR values of the watermarked images are given in Table 1. For all images, it can be seen that the PSNR values are higher than 57 *dB*. Obtained high PSNR watermarked images show that the fidelities of the images are preserved by the proposed watermarking method.

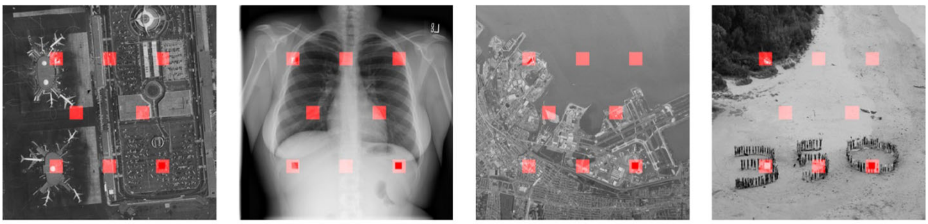
The performance of the proposed tamper detection method is conducted by applying different image processing operations such as histogram equalization (HE), scaling ($x \rightarrow 2x \rightarrow x$), blurring, sharpen, salt and pepper noise addition (SPNA), gaussian noise addition (GNA), average filtering (AF), and cropping to the watermarked images. Random sized attacks are applied to the different regions of the watermarked Lena image, as shown in Fig. 13. By using the proposed tamper detection method, tampered regions are marked in Fig. 14. It can be seen from the figure that detections of the tampered regions have been successfully performed. In order to evaluate the tamper detection method on different images, eight different attacks are also performed to the different regions of size 30×30 of watermarked Boat, Man and Lake images, as shown in Fig. 15. Figure 16 shows the detected tampered regions of the watermarked images.

In order to evaluate the performance of the proposed method for different application areas, 1024×1024 Airport [57], ChestX-ray [56], San Francisco [57] and Threehundred [9] images shown in Fig. 17 are used. The secret key is used in these samples. The images which are watermarked by the proposed method are shown in Fig. 18. The PSNR values of the watermarked images are given in Table 2. Eight different attacks are also performed to the different regions of



(a) (b) (c) (d)

Fig. 19 Implementation of different attacks to the different 30×30 regions of the watermarked images: a) Airport, b) ChestX-ray, c) San Francisco, d) Threehundred



(a) (b) (c) (d)

Fig. 20 Detection of the tampered regions of the watermarked images: a) Airport, b) ChestX-ray, c) San Francisco, d) Threehundred

Table 3 The time complexity of the proposed method

Image Size	Watermark Embedding Time(*sec*)	Tamper Detection Time(*sec*)
256×256	0.481935	0.512542
512×512	1.865400	1.916934
1024×1024	7.328386	7.480778
2048×2048	29.033322	29.978009

size 30×30 of watermarked Airport, ChestX-ray, San Francisco and Threehundred images, as shown in Fig. 19. Figure 20 illustrates the detected tampered regions of the watermarked images.

In order to determine the time complexity of the proposed method, 256×256, 512×512, 1024×1024, 2048×2048 sizes of Lena images are used. The experiments have been performed on a computer with a CPU Intel Core i5-7500 3.40 GHz and 4 GB RAM using the software MATLAB 17b. The total watermark embedding and tamper detection times are given in Table 3. It can be seen from the table that, as the size of the image increases fourfold, the complexity time increases approximately fourfold. Due to the comparison process, the tamper detection time is higher than the watermark embedding time.

5 Conclusions

In this paper, an efficient block based fragile watermarking method has been proposed for image integrity and tamper detection. In watermark embedding phase, host image is divided into 32×32 non-overlapped blocks and block based watermark is generated by using SHA-256 hash function. Then, the generated 256-bit binary watermark is completely embedded into 16×16 sub-block embedding area with LSB modification. In tamper detection phase, the tampered regions of the watermarked image have been detected by comparing the hash value with the extracted watermark. The performance of the method has been demonstrated by applying various attacks to the different regions of the watermarked images. Experimental results illustrate that the high visual quality of watermarked image have been obtained and all the tampered regions have been clearly detected by the proposed method. The most important contribution of our paper is that, by using efficient block based scheme, the quarter part of non-overlapped blocks have been used for watermark embedding area. In addition, generated hash values have been completely used to improve the reliability of our method.

As a future work, different hash functions can be used to improve the proposed method. In addition, in order to detect the tampered pixels rather than tampered blocks, proposed method can be combined with pixel-wise model. Furthermore, the proposed method can be applied to robust watermarking area.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

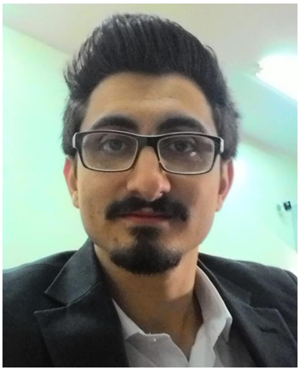
References

1. Ali M, Ahn CW, Pant M (2018) An efficient lossless robust watermarking scheme by integrating redistributed invariant wavelet and fractional fourier transforms. Multimed Tools Appl 77(10):11751–11773

2. Ali SA, Jawad MJ, Naser MA (2017) A semi-fragile watermarking based image authentication. J Eng Appl Sci 12(6):1582–1589
3. Aparna P, Kishore PVV (2018) An efficient medical image watermarking technique in e-healthcare application using hybridization of compression and cryptography algorithm. J Intell Syst 27(1):115–133
4. Aslantas V, Ozer S, Ozturk S (2007) A novel clonal selection algorithm based fragile watermarking method. In: Artificial Immune Systems, Springer, pp 358–369
5. Aslantas V, Dogan AL, Ozturk S (2008) DWT-SVD based image watermarking using particle swarm optimizer. In: Multimedia and Expo, 2008 IEEE International Conference on, IEEE, pp 241–244
6. Aslantas V, Ozer S, Ozturk S (2008) A novel fragile watermarking based on particle swarm optimization. In: Multimedia and Expo, 2008 IEEE International Conference on, IEEE, pp 269–272
7. Aslantas V, Ozer S, Ozturk S (2009) Improving the performance of DCT-based fragile watermarking using intelligent optimization algorithms. Opt Commun 282(14):2806–2817
8. Bravo-Solorio S, Nandi AK (2011) Secure fragile watermarking method for image authentication with improved tampering localisation and self-recovery capabilities. Signal Process 91(4):728–739
9. Christlein V, Riess C, Jordan J, Riess C, Angelopoulou E (2012) An evaluation of popular copy-move forgery detection approaches. arXiv preprint arXiv:12083665
10. Das S, Kundu MK (2013) Effective management of medical information through ROI-lossless fragile image watermarking technique. Comput Methods Prog Biomed 111(3):662–675
11. Fatema M, Maheshkar V, Maheshkar S, Agarwal G (2018) Tamper detection using fragile image watermarking based on chaotic system. In: International Conference on Wireless Intelligent and Distributed Environment for Communication, Springer, pp 1–11
12. Ghosal S, Mandal J (2014) Binomial transform based fragile watermarking for image authentication. J Inf Secur Appl 19(4):272–281
13. Hong W, Chen M, Chen TS (2017) An efficient reversible image authentication method using improved PVO and LSB substitution techniques. Signal Process Image Commun 58:111–122
14. Hsu CS, Tu SF (2010) Probability-based tampering detection scheme for digital images. Opt Commun 283(9):1737–1743
15. Hu YC, Lo CC, Chen WL (2016) Probability-based reversible image authentication scheme for image demosaicking. Futur Gener Comput Syst 62:92–103
16. Khor HL, Liew SC, Zain JM (2017) Region of interest-based tamper detection and lossless recovery watermarking scheme (ROI-DR) on ultrasound medical images. J Digit Imaging 30(3):328–349
17. Kumar C, Singh AK, Kumar P (2018) A recent survey on image watermarking techniques and its application in e-governance. Multimed Tools Appl 77(3):3597–3622
18. Kunhu A, Al-Ahmad H (2013) A new watermarking algorithm for color satellite images using color logos and hash functions. In: 2013 Fifth International Conference on Computational Intelligence, Communication Systems and Networks, pp 251–255
19. Kunhu A, Al-Ahmad H, Mansoori SA (2017) A reversible watermarking scheme for ownership protection and authentication of medical images. In: 2017 International Conference on Electrical and Computing Technologies and Applications (ICECTA), pp 1–4
20. Lee SJ, Jung SH (2001) A survey of watermarking techniques applied to multimedia. In: Industrial Electronics, 2001. Proceedings. ISIE 2001. IEEE International Symposium on, IEEE, vol 1, pp 272–277
21. Li C, Wang Y, Ma B, Zhang Z (2013) Multi-block dependency based fragile watermarking scheme for fingerprint images protection. Multimed Tools Appl 64(3):757–776
22. Li C, Zhang A, Liu Z, Liao L, Huang D (2015) Semi-fragile self-recoverable watermarking algorithm based on wavelet group quantization and double authentication. Multimed Tools Appl 74(23):10581–10604
23. Li CT, Si H (2007) Wavelet-based fragile watermarking scheme for image authentication. J Electron Imaging 16(1):013009
24. Li ZH, Hou JJ (2006) DCT-domain fragile watermarking algorithm based on logistic maps. Acta Electron Sin 34(12):2134
25. Loukhaoukha K, Chouinard JY, Taieb MH (2011) Optimal image watermarking algorithm based on LWT-SVD via multi-objective ant colony optimization. J Inf Hiding Multimed Signal Process 2(4):303–319
26. Lu CS, Liao HYM (2003) Structural digital signature for image authentication: an incidental distortion resistant scheme. IEEE Trans Multimed 5(2):161–173
27. MeenakshiDevi P, Venkatesan M, Duraiswamy K (2009) A fragile watermarking scheme for image authentication with tamper localization using integer wavelet transform. J Comput Sci 5(11):831
28. Navas K, Sasikumar M, Sreevidya S (2007) A benchmark for medical image watermarking. In: Systems, Signals and Image Processing, 2007 and 6th EURASIP Conference focused on Speech and Image Processing, Multimedia Communications and Services. 14th International Workshop on, IEEE, pp 237–240
29. Nazari M, Sharif A, Mollaeefar M (2017) An improved method for digital image fragile watermarking based on chaotic maps. Multimed Tools Appl 76(15):16107–16123

30. Nguyen TS, Chang CC, Yang XQ (2016) A reversible image authentication scheme based on fragile watermarking in discrete wavelet transform domain. AEU - Int J Electron Commun 70(8):1055–1061
31. Ni Z, Shi YQ, Ansari N, Su W (2006) Reversible data hiding. IEEE Trans Circuits Syst Video Technol 16(3):354–362
32. Pandey R, Singh AK, Kumar B, Mohan A (2016) Iris based secure NROI multiple eye image watermarking for teleophthalmology. Multimed Tools Appl 75(22):14381–14397
33. Parekh M, Bidani S, Santhi V (2018) Spatial domain blind watermarking for digital images. In: Progress in Computing, Analytics and Networking, Springer, pp 519–527
34. Patel HA, Divecha NH (2018) A feature-based semi-fragile watermarking algorithm for digital color image authentication using hybrid transform. In: Advances in Computer and Computational Sciences, Springer, pp 455–465
35. Peng Yin HHY (2001) Classification of video tampering methods and countermeasures using digital watermarking
36. Publications (FIPS) FIPS (October 2008) Secure hash standard (shs). Standard, National Institute of Standards and Technology, FIPS PUB 180-3, Gaithersburg, MD 20899–8900
37. Qin C, Chang CC, Chen PY (2012) Self-embedding fragile watermarking with restoration capability based on adaptive bit allocation mechanism. Signal Process 92(4):1137–1150
38. Qin C, Wang H, Zhang X, Sun X (2016) Self-embedding fragile watermarking based on reference-data interleaving and adaptive selection of embedding mode. Inf Sci 373:233–250
39. Qin C, Ji P, Wang J, Chang CC (2017) Fragile image watermarking scheme based on VQ index sharing and self-embedding. Multimed Tools Appl 76(2):2267–2287
40. Qin C, Ji P, Zhang X, Dong J, Wang J (2017) Fragile image watermarking with pixel-wise recovery based on overlapping embedding strategy. Signal Process 138:280–293
41. Singh AK, Dave M, Mohan A (2014) Wavelet based image watermarking: futuristic concepts in information security. Proc Natl Acad Sci India Sect A Phys Sci 84(3):345–359
42. Singh AK, Kumar B, Singh G, Mohan A (2017) Robust and secure multiple watermarking technique for application in tele-ophthalmology. In: Medical Image Watermarking, Springer, pp 159–173
43. Singh D, Singh SK (2017) DCT based efficient fragile watermarking scheme for image authentication and restoration. Multimed Tools Appl 76(1):953–977
44. Singh P, Agarwal S (2017) A self recoverable dual watermarking scheme for copyright protection and integrity verification. Multimed Tools Appl 76(5):6389–6428
45. Singh P, Chadha R (2013) A survey of digital watermarking techniques, applications and attacks. Int J Eng Innov Technol (IJEIT) 2(9):165–175
46. Singh RK, Shaw DK, Alam MJ (2015) Experimental studies of LSB watermarking with different noise. Procedia Comput Sci 54:612–620
47. Sreenivas K, Kamkshi Prasad V (2017) Fragile watermarking schemes for image authentication: a survey. Int J Mach Learn Cybern
48. Su Q, Chen B (2018) Robust color image watermarking technique in the spatial domain. Soft Comput 22(1):91–106
49. Su Q, Wang G, Lv G, Zhang X, Deng G, Chen B (2017) A novel blind color image watermarking based on contourlet transform and hessenberg decomposition. Multimed Tools Appl 76(6):8781–8801
50. Trivedy S, Pal AK (2017) A logistic map-based fragile watermarking scheme of digital images with tamper detection. Iran J Sci Technol Trans Electr Eng 41(2):103–113
51. Tsai P, Hu YC, Chang CC (2005) Novel image authentication scheme based on quadtree segmentation. Imaging Sci J 53(3):149–162
52. Ustubioglu A, Ulutas G (2017) A new medical image watermarking technique with finer tamper localization. J Digit Imaging 30(6):665–680
53. Vasu S, George SN, Deepthi PP (2012) An integrity verification system for images using hashing and watermarking. In: 2012 International Conference on Communication Systems and Network Technologies, pp 85–89
54. Verma VS, Jha RK, Ojha A (2015) Significant region based robust watermarking scheme in lifting wavelet transform domain. Expert Syst Appl 42(21):8184–8197
55. Wahid M, Ahmad N, Zafar MH, Khan S (2018) On combining MD5 for image authentication using LSB substitution in selected pixels. In: Engineering and Emerging Technologies (ICEET), 2018 International Conference on, IEEE, pp 1–6
56. Wang X, Peng Y, Lu L, Lu Z, Bagheri M, Summers RM (2017) Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In: Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on, IEEE, pp 3462–3471
57. Weber AG (1997) The USC-SIPI image database version 5. USC-SIPI Rep 315:1–24

58. Yu M, Wang J, Jiang G, Peng Z, Shao F, Luo T (2015) New fragile watermarking method for stereo image authentication with localization and recovery. AEU - Int J Electron Commun 69(1):361–370
59. Zhang H, Wang C, Zhou X (2017) Fragile watermarking based on LBP for blind tamper detection in images. JIPS (J Inf Process Syst) 13(2):385–399
60. Zhang X, Wang S (2009) Fragile watermarking scheme using a hierarchical mechanism. Signal Process 89(4):675–679
61. Zhang X, Xiao Y, Zhao Z (2015) Self-embedding fragile watermarking based on DCT and fast fractal coding. Multimed Tools Appl 74(15):5767–5786
62. Zheng PP, Feng J, Li Z, Zhou MQ (2014) A novel SVD and LS-SVM combination algorithm for blind watermarking. Neurocomputing 142:520–528



Ertugrul Gul received his BSc degree in Computer Engineering in 2015, MSc degree in Computer Engineering in 2017, from Erciyes University, Kayseri, Turkey. He is a PhD student in Computer Engineering from Erciyes University, Kayseri, Turkey and currently working as a research assistant in Computer Engineering Department at the Erciyes University. His research interests include information security, image processing.

Serkan Ozturk received his BSc degree in Computer Engineering from Karadeniz Technical University, Trabzon, Turkey, in 1999, MSc degree in Electronic Engineering in 2002, and PhD degree in Electronic Engineering from Erciyes University, Kayseri, Turkey, in 2009. He conducted a post-doctoral research in the Department of Computer Science at Ryerson University, Toronto, Canada, between 2010 and 2011. He is currently working as an assistant professor in the Department of Computer Engineering at the Erciyes University. His research interests include information security, image processing, and wireless communication.