A Feature-based Fragile Watermarking for Tamper Detection using Voronoi Diagram Decomposition

Nour El-Houda GOLEA[1] and Kamal Eddine MELKEMI[2]

[1] Department of Computer Science, University of Batna2, Algeria
nh.golea@univ-batna2.dz
[2] Department of Computer Science, University of Batna2, Algeria
melkemi2002@yahoo.com

Abstract. In this paper, we have proposed a novel feature-based fragile watermarking scheme for image authentication. The proposed technique extracts *Feature Points* (FP) by performing the Harris corner detector and used them as germs to decomposes the host image in segments using *Voronoi Diagram* (VD). The authentication of each segment is guaranteed by using the *Cyclic Redundancy Check code* (CRC). Then, the CRC encoding procedure is applied to each segment to generate the watermark. Voronoi decomposition is employed because it has a good retrieval performance compared to similar geometrical decomposition algorithms. The security aspect of our proposed method is achieved by using the public key crypto-system RSA (Rivest–Shamir–Adleman) to encrypt the FP.
Experimental results demonstrate the efficiency of our approach in terms of imperceptibility, the capability of detection of alterations, the capacity of embedding, and computation time. We have also prove the impact of VD decomposition on the quality of the watermarked image compared to block decomposition.
The proposed method can be applicable in the case where the tamper detection is critical and only some regions of interest must be re-transmitted if they are corrupted, like in the case of medical images. An example of the application of our approach to medical image is briefly presented.

Keywords: Fragile watermarking, feature-based image watermarking, image authentication, Voronoi diagram (VD).

1 Introduction

To protect a digital image, a pattern of bits can be inserted into this image for identifying its copyright information. This approach is called *digital watermarking*[1].

According to the emebedding domain, the watermark information can be embedded in three different ways: directly in pixels of original image (*spatial domain*) [21], in coefficients of image transformation (*transform domain*) [3] or in features of image (*feature domain*)[17]. Feature-based watermarking schemes are based on salient regions, Feature Points (FP) or image characteristics, and are efficient in terms of detection and recovery against geometric attacks. Several proposed FP-based watermarking approaches used Harris corner detector [2], [29], Harris-Laplacian [10], Mexican hat wavelet [5] or scale invariant feature transform SIFT [18].

David C. Wyld et al. (Eds): CCSEA, BIoT, DKMP, CLOUD, NLCAI, SIPRO - 2020
pp. 141-160, 2020. CS & IT - CSCP 2020 DOI: 10.5121/csit.2020.101012

The watermarking technique has several applications as copyright protection, image authentication, and integrity and according to these applications, the watermarking can be respectively: robust [21], fragile [3] and semi-fragile [11]. Fragile watermarking must be sensitive both to malicious attacks and to accidental content alteration and it is very desirable to detect corrupted areas.

An efficient fragile watermarking technique must insure some proprieties [12]. First, it must embed the watermark imperceptibility, which means that the watermark should be embedded into the host media invisibly. Second, the watermarking technique must be robust to malicious attacks that try to damage the watermark functionality. Third, the tampering should be detected without using the original image. Finally, the fragile watermarking technique must be able to locate the tampered regions within an image. There are three kinds of location accuracy, named: locating single-pixel tampered, locating single-block tampered and locating the whole signal.

The concept of error checking and correcting codes is widely used in the design of fragile watermarking systems. Lin and al.[15] proposed a fragile block-wise and content-based watermarking for image authentication and recovery. In this scheme, the watermark of each block is an encrypted form of its signature, which includes the block location, a content-feature of another block, and a CRC checksum. Where the CRC checksum is used to authenticating only the signature. In [8], a pixel-wise fragile watermarking approach is proposed. This approach authenticates an RGB image using the CRC checksum. However, the degree of the generator polynomial is very small and does not go beyond six. The most important parameter in error detection of a message stream is the selection of the generator polynomial. To overcome this insufficiency and enlarge the degree of the generator polynomial, we propose a region-wise fragile watermarking technique using a standard polynomial generator $G(x)$ having particular mathematical properties like CRC-32, CRC-16, and CRC-8 to generate the watermark. The variation of the size of the polynomial allows on one hand to increase the authentication guarantee and to strengthen the security aspect on the other hand. In another work [9], the standard CRC-32 is used to authenticate only the Region of interest of the medical image. To achieve the authentication and integrity of different types of images, we propose a novel CRC-based fragile watermarking scheme based on FP and VD decomposition of the image.

We have organized this paper as follows: Section 2 presents a state of the art on feature based watermarking approaches. Section 3 gives a description of VD. Our approach is presented in Section 4. In Section 5, the experimental results are reported and analyzed. Finally, we concluded our work in Section 6.

2 Related works

In [4], Bhattacharjee et al. propose a semi-fragile approach for authenticating digital images using a Mexican hat wavelet to extract FP. These features are defined to be relatively unaffected by lossy compression.

Kutter et al. [13], propose a robust image watermarking scheme. They use the Mexican Hat Wavelet to extract features points and the Voronoi diagram (VD) to define watermark embedding regions. The watermark in each segment is centered on the location of the corresponding feature. In the extraction process, the same FP are detected and used to partition the image. Then the watermark is extracted from each partition. The drawback of this approach is that the location of FP may be changed by some pixels because of attack or during the watermarking process. This change will cause problems during the detecting process.

Bas et al. [2], developed another watermarking scheme using the Harris detector to extract feature points. Then, performing Delaunay Tessellation on the set of features to construct a triangular tessellation that they use to embed the watermark.

In [23], Tang and Hang propose a robust image watermarking method which adopts the Mexican Hat wavelet scale interaction to extract feature points. These feature points are employed as centers of disks that are watermarked. For each disk, two blocks are selected by using the image normalization technique, and the watermark is embedded in the DFT domain of the two blocks of each disk separately.

In [22] a robust watermarking algorithm based on the discrete cosine transform (DCT) and Voronoi Diagram to segment the image is proposed. The feature extraction point which is used to form the Voronoi segment is built based on Tommasini et al. algorithm [24]. So, for each segment, the image segment is subdivided into blocks of size 8×8, (64 pixels). The DCT of the block is then computed. After that, the DCT coefficients are re-ordered into a zigzag scan. A pseudo-random sequence of real numbers is embedded in the DCT coefficients. Therefore, the modified DCT coefficients are re-inserted in the zigzag scan. Then, the inverse DCT is applied. Finally, the blocks are merged. Thus, we can obtain the watermarked image after merging all image segments.

Seo et al.[19], propose to utilize the Harris-Laplacian method. Scale Invariant Feature Points are detected based on the scale selection at Harris corner points. In the spatial domain, the watermark is embedded in a circularly symmetric way centered at each selected feature point. In the detection stage, the feature points are found similarly and the existence of the watermark decided with correlation enhanced with SPOMF (Symmetrical Phase Only Filter).

Su et al. [20], apply segmentation to determine feature-based spatially localized structures for watermark embedding and detection. This method offers good tolerance to collusion attacks and reasonable robustness to geometric distortions.

Lee et al. [14], developed a watermarking method that is robust to geometric distortions. They use the Scale Invariant feature transform (SIFT) to extract image

feature points, which are used to generate the number of circular regions. The watermark is inserted into the circular patches in an additive way in the spatial domain. Rotation invariance is achieved using the translation property of the polar-mapped circular patches.

Qi et al.[28] develop a robust content-based watermarking scheme. The image content is represented by important feature points obtained by the image-texture-based adaptive Harris corner detector. These important feature points are geometrically significant and therefore are capable of determining the possible geometric attacks with the aid of the Delaunay-tessellation-based triangle matching method.

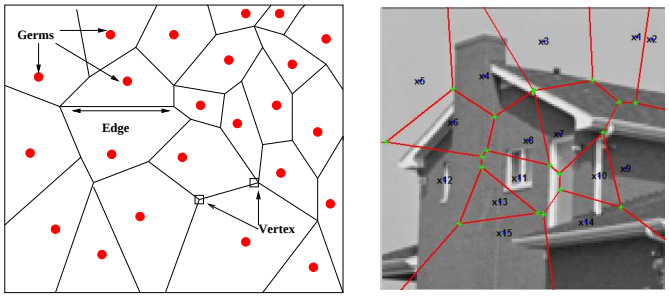
In [25] Wang et al. proposed another feature-based image watermarking scheme robust to general geometric attacks. The Harris-Laplace detector is also utilized to extract steady feature points from the host image. Then, the local feature regions (LFR) are ascertained adaptively according to the characteristic scale theory, and they are normalized by an image normalization technique. Finally, according to the pre-distortion compensation theory, several copies of the digital watermark are embedded into the non-overlapped normalized LFR by comparing the DFT mid-frequency magnitudes.

Wei et al. [27], present a robust watermarking scheme based on feature point detection and image normalization. Feature points are detected from the original image using the proposed multiresolution feature point detection filter. Then, image normalization is applied to the disks centered at these feature points. The watermark is embedded in the subband coefficients of the DFT domain of each disk separately. And Watermark detection is based on a local threshold on normalized correlation to detect if a disk has been watermarked, as well as on a global threshold to detect if the image has been watermarked.

Yuan et al.[29], propose a robust geometric invariant digital image watermarking scheme based on feature extraction and local Zernike moments. The features points are extracted employing the Adaptive Harris Detector. Each extracted circular patch is decomposed into a collection of binary images and Zernike transform is applied to the selected binary patches. A spread spectrum communication technique is used to embed the watermark. After the watermark is embedded, the inverse Zernike transform is applied to reconstruct the corresponding binary patch from the watermarked Zernike moments. Finally, the watermarked image can be obtained by replacing the original patches with the watermarked patches. For watermark extraction, the inverse procedure of watermark embedding is operated. The linear correlation is used to detect the existence of the watermark in the Zernike moments magnitudes. The watermark is detected when the linear correlation result is larger than a predefined threshold value.

3 Voronoi diagram

VD is defined as a partition of the plane into polygons or regions according to the principle of the nearest neighbor. Given a set of $2D$ points $P = \{p_1, p_2, \ldots, p_n\}$. The Voronoi region for a point p_i is defined as the set of all the points that are closer to p_i than to any other points. Points p_i are called *Voronoi generators* or *germs*. The edge common to two Voronoi regions (VR) is called a *Voronoi edge*. The vertices where three or more Voronoi edges meet are named *Voronoi vertices*. We say that a Voronoi generator p_i is adjacent to p_j when their Voronoi regions share a common edge [26]. Figure 1 illustrates VD concepts and decomposed *House* image using VD.



(a) (b)

Fig. 1. (a) VD concepts. (b) decomposed *House* image using VD.

4 Proposed approach

The proposed method is decomposed from three algorithms: watermark generator, watermark embedding, and verification.

For further details, the watermark is generated by performing the Algorithm 1 according to the following steps: First, the Harris corner detector is used to extract FP. From these FP considered as Voronoi germs, the image is decomposed using VD, where each Voronoi polygon is the message to be transmitted. Second, pixels of each message are decomposed in packets of normalized size according to the degree r of generator polynomial $G^r(x)$. For example, using $G^{32}(x)$ the number of pixels in the packet is 16 because 2 bits LSB (*Least Significant Bit*) from each pixel are used to embed the checksum of size 32. The six bits MSB (*Most Significant Bit*) of each pixel are extracted and concatenated to create a sequence of normalized size. We perform the CRC Encoder at this sequence using a $G^r(x)$. Every two bits of the calculated checksum are inserted into two LSB of the corresponding pixel using Algorithm 2. At the detection and verification process (Algorithm 3), the

receiver uses the same FP to create the Voronoi cells and sub-device the pixels in packets of normalized size. Two LSB bits of each pixel are extracted and concatenated to create the extracted checksum. A new sequence is created by appending the checksum at the end of the concatenated six bits MSB of each pixel. On dividing an error-free sequence with an m degree generator polynomial, the remainder polynomial of degree $m-1$ should be all 0's. The remaining non-zero combinations of the remainder polynomial directly specify the error polynomials each of which identifies the single-bit error position. According to the importance and quantity of the alteration in the region, the receiver requests from the sender only the retransmission of the altered packets. The security aspect of our proposed method is achieved by using the public key crypto-system RSA to encrypt the FP.

Algorithm 1 Watermark generation

Input: f: original image.
Output: W: watermark, FP_{Cryp}: a set of encrypted feature points.
Steps:

1. Select a set of N features points $FP = \{p_1, p_2, \ldots, p_N\}$ using Harris Corner Detector and encrypted them using RSA to generate FP_{Cryp}.
2. Decompose f by creating N Voronoi regions (VR) using FP as germs. Each region $VR(p_i)$ is considered as a message to be transmitted.
3. For each message $VR(p_i)$ do :
   - Divide the message $VR(p_i)$ on X packets of size $S_j = \{16, 8, 4, 2 \text{ or } 1\}$, where j is the number of packet. For example, if the first region $VR(p_1)$ is composed from 47 pixels, $VR(p_1) = \{f_1, \ldots, f_{47}\}$. The X packets are :
     - $X_1 = \{f_1, \ldots, f_{16}\}$ of size $S_1 = 16 \Rightarrow G(x)$ of degree 32;
     - $X_2 = \{f_{17}, \ldots, f_{32}\}$ of size $S_2 = 16 \Rightarrow G(x)$ of degree 32;
     - $X_3 = \{f_{33}, \ldots, f_{40}\}$ of size $S_3 = 8 \Rightarrow G(x)$ of degree 16;
     - $X_4 = \{f_{41}, \ldots, f_{44}\}$ of size $S_4 = 4 \Rightarrow G(x)$ of degree 8;
     - $X_5 = \{f_{45}, f_{46}\}$ of size $S_5 = 2 \Rightarrow G(x)$ of degree 4;
     - $X_6 = \{f_{47}\}$ of size $S_6 = 1 \Rightarrow G(x)$ of degree 2;
   - For each packet X_j of size S_j do :
     - Extract the six bits MSB of each pixel and concatenate them together to create a sequence m.
     - Appended $2 \times S_j$ zeros bits at the end of m to create m'. This is equivalent to calculate $m' = m \times x^{2 \times S_j}$
     - The watermark W_j^i is the remainder of division of m' by the normalized generator of degree $d = 2 \times S_j$.

Algorithm 2 Watermark embedding

Input: f: original image. W: watermark.
Output: f_w: watermarked image.
Steps:

1. Decompose f by creating N Voronoi Regions (VR) using FP as germs.
2. For each message $VR(p_i)$ do :
   - Divide the message on X packets of size $S_j = \{16, 8, 4, 2 \text{ or } 1\}$, where j is the number of the packet.
   - For each packet X_j of size S_j do :
     - Insert every two bits of the corresponding watermark W_j^i in the two bits LSB of each pixel.
   - Reconstruct the watermarked packet $VR_W(p_i)$ using the watermarked pixels.
   - Rearrange the X watermarked packets to reconstruct the watermarked message $VR_w(p_i)$.
3. Rearrange the N watermarked messages (segments) VR_W to reconstruct the watermarked image f_w.

Algorithm 3 Verification

Input: f_w^*: possibly distorted watermarked image. FP_{Cryp}: the set of encrypted feature points.
Output: VM : Verification Map.
Steps:

1. Decrypt FP_{Cryp}.
2. Create N watermarked Voronoi region (VR_w) using decrypted FP as germs.
3. For each message $VR_w(p_i)$ do :
   - Divide each watermarked message $VR_w(p_i)$ on X^* packets of pixels of size $S_j = \{16, 8, 4, 2 \text{ or } 1\}$.
   - For each watermarked packet X_j^* of size S_j do :
     - Extract the two bits LSB of each pixel and concatenate these bits together to create the extracted checksum $W_j^{i^*}$.
     - Extract the six bits MSB of each pixel and concatenate these bits together to create a sequence m^*.
     - Appended $W_j^{i^*}$ at the end of m^* to create m_w^*.
     - Divide m_w^* by the normalized CRC of the degree $d = 2 \times S_j$.
     - If the remainder of the division is not zero then the packet X_j^* is corrupted.
   - If one of X^* packets is corrupted then the message $VR_w(p_i)$ is also corrupted .

Demonstrative schemes of the generator, embedding, and verification algorithms are respectively shown in Figures 2 ,3, and 4.

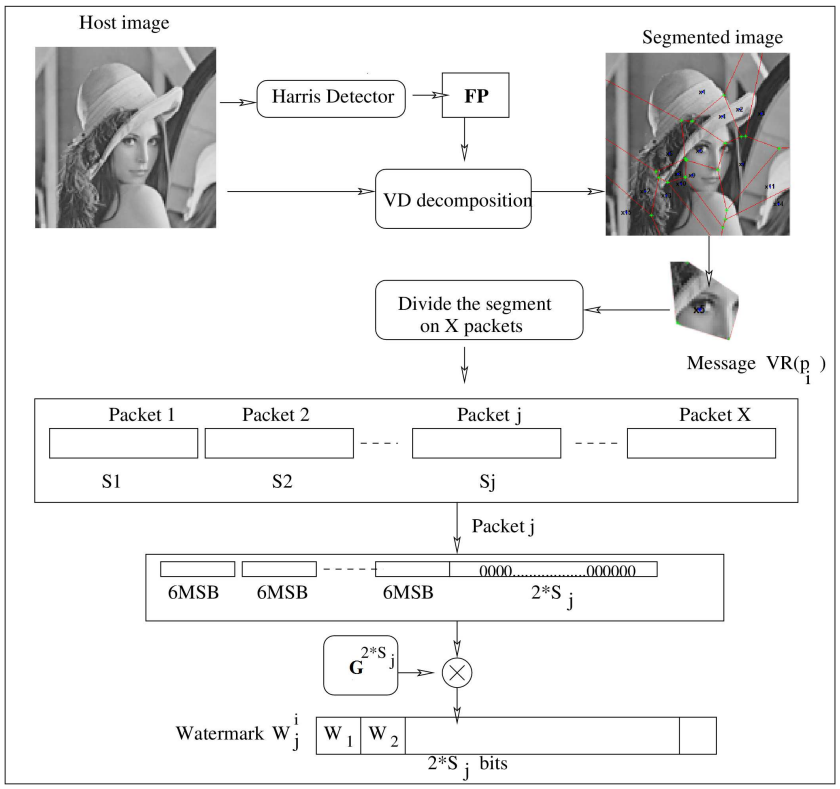


Fig. 2. Watermark generation.

5 Experimental results

In this section, some preliminary experiments have been carried out to evaluate the effectiveness of our watermarking scheme. These tests are based on imperceptibility, fragility, capacity, computational time, and the impact of the decomposition on the quality. We have performed our watermark embedding on several grayscale images with different sizes and compared our scheme with a similar fragile watermarking method [6].

5.1 Imperceptibility analysis

Several typical grayscale images with different sizes have been watermarked, in order to assess the imperceptibility property of our watermarking method. Original images and their watermarked ones have, respectively, been shown in Figure 5. From these result images, we could see that the distortion after the embedding proceed is hard to be perceived by human eyes.

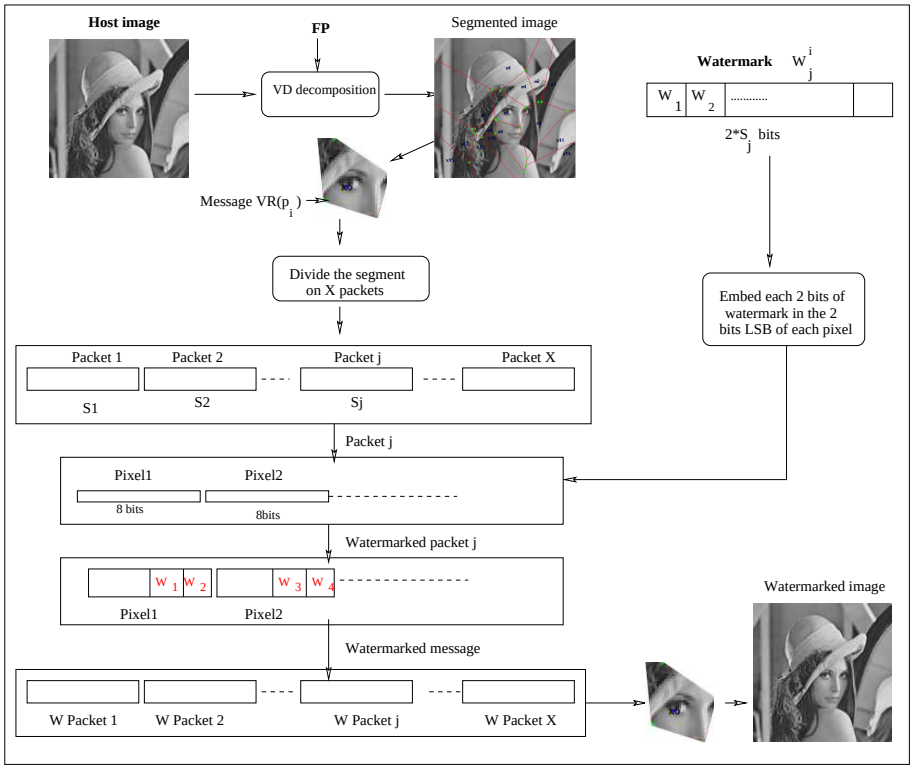


Fig. 3. Watermark embedding.

Two metrics are used to evaluate the imperceptibility propriety, PSNR (Peak Signal to Noise Ratio), and SSIM (Structural Similarity Metric Index).

The PSNR and SSIM values for different sizes of host images obtained by our scheme compared to Singh's scheme [6] have illustrated in Table 1. The results reported in Table 1 show satisfactory results of the proposed scheme. In every case, the PSNR values are higher than 47 dB and are best than the method of Singh [6]. The size of the host image in Singh's scheme [6] is limited to 256×256 because the position of the pixel (column and row) are converted into 8 bits binary representation.

5.2 Fragility analysis

To test the capability of the watermarking approach to detect the attack is similar to evaluate your capability to detect the errors. At this stage, we study several scenarios of altered bits in the watermarked pixels. In addition to the comparison of our method with Singh's scheme [6], we have also compared the proposed scheme with the method using the polynomial generator of degree 3 (CRC-3) at each five

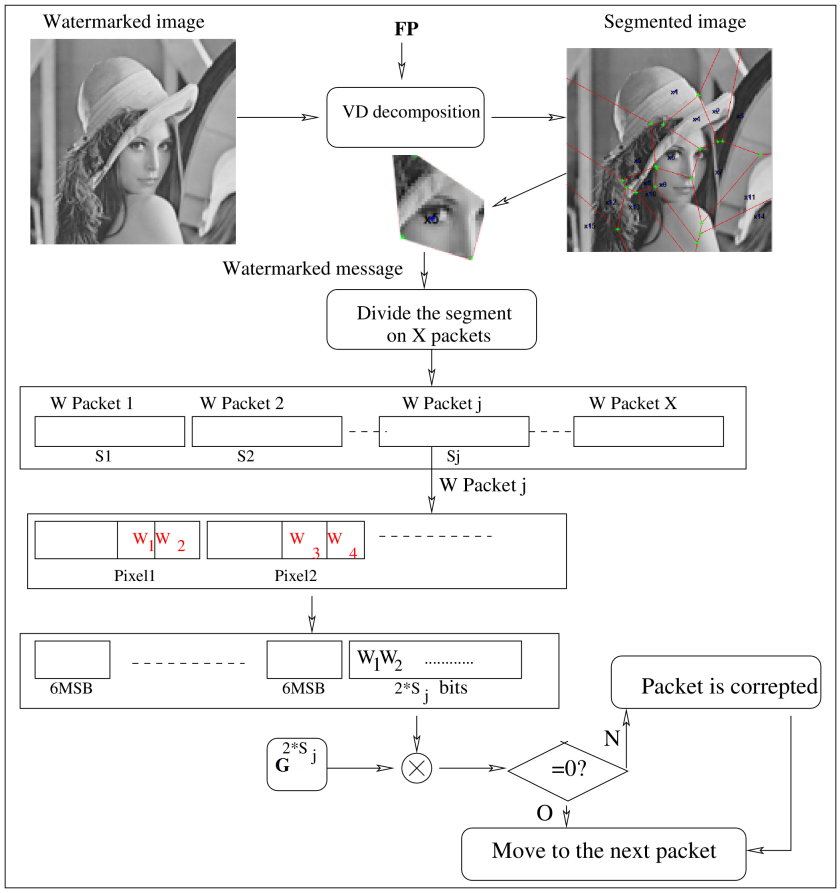


Fig. 4. Watermark verification.

	Proposed approach			Singh's approach	
Host image	Size	PSNR	SSIM	PSNR	SSIM
Air-plane	64×64	47.04	0.978	41.51	0.970
Baboon	128×128	47.19	0.994	41.02	0.978
Elaine	256×256	47.12	0.985	41.02	0.948
Lena	512×512	47.15	0.980	-	-

Table 1. The PSNR and SSIM values for different sizes of host images.

bits MSB. So, the generated watermark is inserted directly in the three LSB like in Singh's method. So, we have appended three zeros bits at the end of the five bits MSB of each pixel. This sequence is divided by a generator polynomial of degree 3 (CRC-3). The remainder of the division is inserted at the three bits LSB of each pixel. In the verification process, the receiver appended the three bits LSB

Airplane 64×64 | Baboon 128×128 | Elaine 256×256 | Lena 512×512

Fig. 5. Original and watermarked images.

at the end of the five MSB bits. This sequence is divided by CRC-3. The pixel is not corrupted if the remainder equals zero, else it is corrupted. Table 2 illustrates different scenarios of altered bits and the capability of watermarking approaches to detect errors in the same watermarked pixel.

It's clear from Table 2 that by applying the CRC-3 based method only three errors from 34 errors are not detected. By using Singh's approach 20 errors are not detected. However, our scheme detects all errors.

To visually estimate the fragility of the proposed scheme, we use Verification Map (VM) image to indicate the corrupted packets in each region. If there is no attack, VM is a black image, else white pixels indicate the corrupted pixels. To highlight the fragility of our method, we have taken into account several kinds of image watermarking attacks. Figure 6 shows the attacked Lena images with size 512×512 and their corresponding extracted Verification Map.

5.3 Capacity analysis

The capacity of the watermarking scheme on $N \times M$ image is calculated as follows:

$$Capacity = N \times M \times \frac{N_w}{8} \times 100. \tag{1}$$

Nw: is the number of watermarked bits.

In our scheme $N_w = 2$, the capacity is 25% of the size of the host image. This capacity can be considered high. In [6] scheme, $N_w = 3$ and the capacity is 37.5% of the size of the host image. This capacity is better than the capacity achieved by our scheme, but in [6] the size of the host image is limited to 256×256.

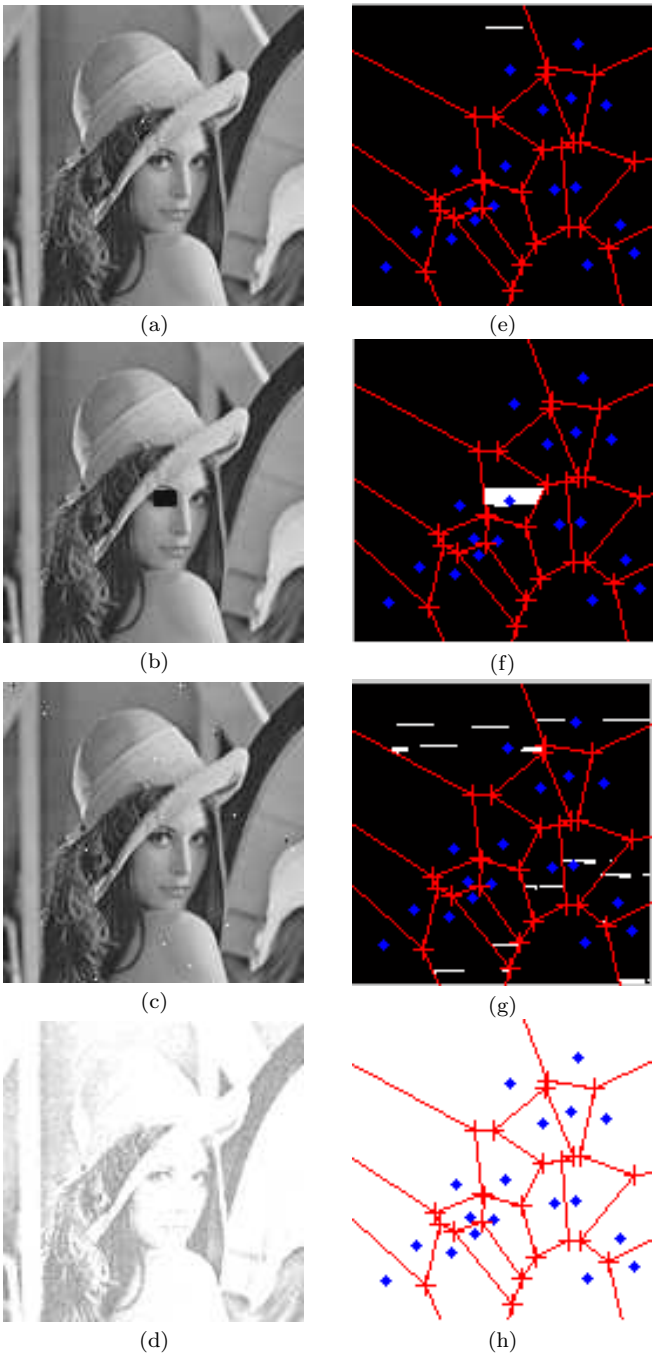


Fig. 6. Fragility against attacks: (a) One pixel corrupted. (b) Cropping attacks. (c) Salt and peppers noise. (d) Gaussian noise. (e-h) Verification Map: detected corrupted packets in each region after different attacks.

	Singh's approach[6]		CRC-3 approach		Our approach	
Original pixel	1 0 0 1 1 0 1 1	155	1 0 0 1 1 0 1 1	155	1 0 0 1 1 0 1 1	155
Watermarked pixel	1 0 0 1 1 0 1 0	154	1 0 0 1 1 0 0 1	153	1 0 0 1 1 0 0 0	152
Altered bits						
LSB bits						
1	1 0 0 1 1 0 1 **1**	155 -	1 0 0 1 1 0 0 **0**	152 +	1 0 0 1 1 0 0 **1**	153 +
2	1 0 0 1 1 0 **0** 0	152 +	1 0 0 1 1 0 **1** 1	155 +	1 0 0 1 1 0 **1** 0	154 +
3	1 0 0 1 1 **1** 1 0	158 -	1 0 0 1 1 **1** 0 1	157 +	1 0 0 1 1 **1** 0 0	156 +
1-2	1 0 0 1 1 0 **0** **1**	153 +	1 0 0 1 1 0 **1** **0**	154 +	1 0 0 1 1 0 **1** **1**	155 +
1-3	1 0 0 1 1 **1** 1 **1**	159 -	1 0 0 1 1 **1** 0 **0**	156 +	1 0 0 1 1 **1** 0 **1**	157 +
2-3	1 0 0 1 1 **1** **0** 0	156 +	1 0 0 1 1 **1** **1** 1	157 +	1 0 0 1 1 **1** **1** 0	158 +
1-2-3	1 0 0 1 1 **1** **1** **1**	157 +	1 0 0 1 1 **1** **0** **1**	158 +	1 0 0 1 1 **1** **1** **1**	159 +
MSB bits						
4	1 0 0 1 **0** 0 1 0	146 +	1 0 0 1 **0** 0 0 1	145 +	1 0 0 1 **0** 0 0 0	144 +
5	1 0 0 **0** 1 0 1 0	138 -	1 0 0 **0** 1 0 0 1	137 +	1 0 0 **0** 1 0 0 0	138 +
6	1 0 **1** 1 1 0 1 0	186 -	1 0 **1** 1 1 0 0 1	185 +	1 0 **1** 1 1 0 0 0	184 +
7	1 **1** 0 1 1 0 1 0	218 +	1 **1** 0 1 1 0 0 1	117 +	1 **1** 0 1 1 0 0 0	216 +
8	**0** 0 0 1 1 0 1 0	26 -	**0** 0 0 1 1 0 0 1	25 +	**0** 0 0 1 1 0 0 0	24 +
8-7	**0** **1** 0 1 1 0 1 0	90 -	**0** **1** 0 1 1 0 0 1	89 +	**0** **1** 0 1 1 0 0 0	88 +
8-6	**0** 0 **1** 1 1 0 1 0	58 +	**0** 0 **1** 1 1 0 0 1	57 +	**0** 0 **1** 1 1 0 0 0	56 +
8-5	**0** 0 0 **0** 1 0 1 0	10 +	**0** 0 0 **0** 1 0 0 1	9 +	**0** 0 0 **0** 1 0 0 0	8 +
8-4	**0** 0 0 1 **0** 0 1 0	18 -	**0** 0 0 1 **0** 0 0 1	17 -	**0** 0 0 1 **0** 0 0 0	16 +
7-6	1 **1** **1** 1 1 0 1 0	250 -	1 **1** **1** 1 1 0 0 1	249 +	1 **1** **1** 1 1 0 0	248 +
7-5	1 **1** 0 **0** 1 0 1 0	202 -	1 **1** 0 **0** 1 0 0 1	201 +	1 **1** 0 **0** 1 0 0 0	200 +
7-4	1 **1** 0 1 **0** 0 1 0	210 +	1 **1** 0 1 **0** 0 0 1	209 +	1 **1** 0 1 **0** 0 0 0	208 +
6-5	1 0 **1** **0** 1 0 1 0	170 +	1 0 **1** **0** 1 0 0 1	169 +	1 0 **1** **0** 1 0 0 0	168 +
6-4	1 0 **1** 1 **0** 0 1 0	178 -	1 0 **1** 1 **0** 0 0 1	177 +	1 0 **1** 1 **0** 0 0 0	176 +
5-4	1 0 0 **0** **0** 0 1 0	130 -	1 0 0 **0** **0** 0 0 1	129 +	1 0 0 **0** **0** 0 0 0	128 +
8-7-6	**0** **1** **1** 1 1 0 1 0	122 -	**0** **1** **1** 1 1 0 0 1	121 +	**0** **1** **1** 1 1 0 0 0	120 +
8-7-5	**0** **1** 0 **0** 1 0 1 0	74 -	**0** **1** 0 **0** 1 0 0 1	73 +	**0** **1** 0 **0** 1 0 0 0	72 +
8-7-4	**0** **1** 0 1 **0** 0 1 0	82 +	**0** **1** 0 1 **0** 0 0 1	81 +	**0** **1** 0 1 **0** 0 0 0	80 +
7-6-5	1 **1** **1** **0** 1 0 1 0	234 -	1 **1** **1** **0** 1 0 0 1	233 +	1 **1** **1** **0** 1 0 0 0	232 +
7-6-4	1 **1** **1** 1 **0** 0 1 0	242 +	1 **1** **1** 1 **0** 0 0 1	241 +	1 **1** **1** 1 **0** 0 0 0	240 +
6-5-4	1 0 **1** **0** **0** 0 1 0	162 -	1 0 **1** **0** **0** 0 0 1	161 +	1 0 **1** **0** **0** 0 0 0	160 +
8-7-6-5	**0** **1** **1** **0** 1 0 1 0	106 +	**0** **1** **1** **0** 1 0 0 1	105 -	**0** **1** **1** **0** 1 0 0 0	104 +
8-7-6-4	**0** **1** **1** 1 **0** 0 1 0	114 -	**0** **1** **1** 1 **0** 0 0 1	113 +	**0** **1** **1** 1 **0** 0 0 0	112 +
8-7-5-4	**0** **1** 0 **0** **0** 0 1 0	66 -	**0** **1** 0 **0** **0** 0 0 1	65 +	**0** **1** 0 **0** **0** 0 0 0	64 +
8-6-5-4	**0** 0 **1** **0** **0** 0 1 0	34 +	**0** 0 **1** **0** **0** 0 0 1	33 +	**0** 0 **1** **0** **0** 0 0 0	32 +
7-6-5-4	1 **1** **1** **0** **0** 0 1 0	226 -	1 **1** **1** **0** **0** 0 0 1	225 -	1 **1** **1** **0** **0** 0 0 0	224 +
8-7-6-5-4	**0** **1** **1** **0** **0** 0 1 0	98 -	**0** **1** **1** **0** **0** 0 0 1	97 +	**0** **1** **1** **0** **0** 0 0 0	96 +

(+): error is detected . (-): error is not detected .

Table 2. Scenarios of altered bits and the capability of watermarking methods to detect the errors.

5.4 Computational time analysis

We also analyzed the time complexity of the proposed scheme to investigate its computational efficiency. In our experiments, a laptop computer with an Intel i3 CPU 2.GHZ, 4 GB RAM, Windows 7 is used as the computing platform.

The experimental results are given in Table 3. Figure 7 presents the effects of increasing image size on execution time performing on Airplane image (Total time= Time Embedding + Time Verification, and Time Embedding = Time generating the watermark + Time including the watermark).

From these results, we can see that our proposed method offers faster computation compared to Singh's method [6].

		Our approach		Singh's approach[6]	
Host image	size	Embedding (s)	Verification (s)	Embedding (s)	Verification (s)
Air-plane					
	64×64	2.83	1.30	61.52	63.044
	128×128	7.86	4.37	245.59	246.60
	256×256	22.8	14.28	990.89	973.00
Baboon					
	64×64	1.89	0.83	73.93	47.86
	128×128	6.42	4.25	159.54	243.96
	256×256	17.20	41.00	976.73	932.78
Elaine					
	64×64	1.71	0.68	57.71	58.89
	128×128	7.24	3.50	229.74	231.16
	256×256	17.20	41.00	953.47	910.63
Lena					
	64×64	2.66	1.23	59.54	57.73
	128×128	8.04	3.77	232.46	232.17
	256×256	30.61	12.31	928.92	919.69
Man					
	64×64	2.36	1.31	57.03	57.06
	128×128	8.78	3.50	229.43	228.71
	256×256	26.71	20.42	919.18	963.27
Peppers					
	64×64	2.70	1.21	64.41	57.79
	128×128	7.18	3.84	254.88	241.37
	256×256	20.95	11.52	969.69	933.57
Splash					
	64×64	1.46	0.54	56.99	57.02
	128×128	6.43	4.03	254.43	228.29
	256×256	10.76	5.52	971.03	940.86
Tree					
	64×64	6.474245	1.932281	56.19	57.19
	128×128	8.46	5.521817	242.48	216.83
	256×256	23.857	14.00	739.05	755.66

Table 3. Computational time.

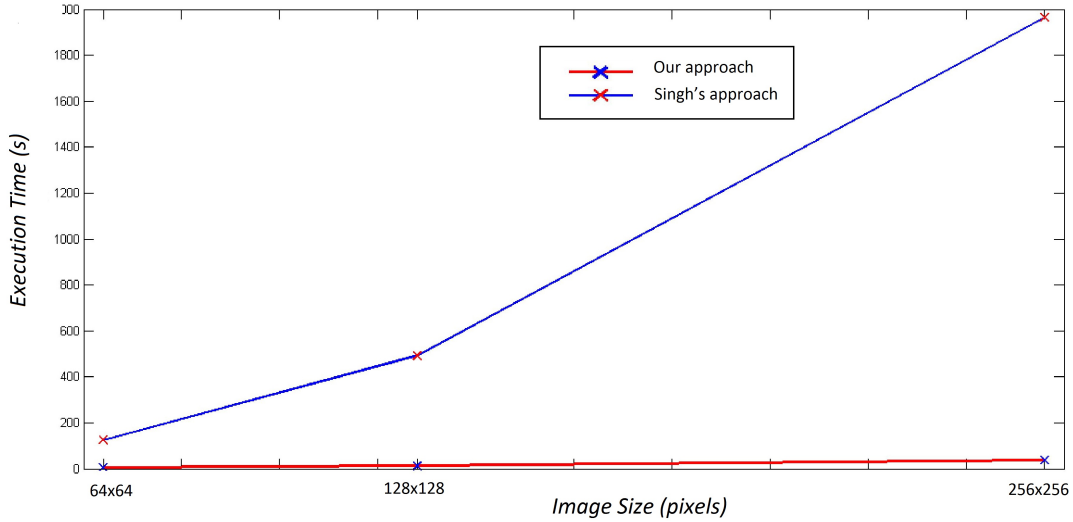


Fig. 7. The effects of increasing image size on the execution time for *Airplane* image.

5.5 The impact of the image decomposition approach

We have verified the superiority of VD decomposition compared to block decomposition. Indeed, we have applied a similar proposed algorithm using block decomposition instead Voronoi decomposition. In fact, the host image is decomposed on blocks of size 16×16. For each block, the MSB bits for each pixel are concatenated to create a sequence m. A new sequence m' is created by appending 2×16 zeros bits at the end of m. The watermark of the corresponding block is the remainder of the division of m' by CRC-32. Finally, embedding every two bits of the watermark on the two bits LSB. In the verification process, the receiver decomposes the image on blocks of size 16×16. For each block, the two bits LSB of each pixel are extracted and concatenated to create the extracted checksum. This checksum is appended at the end of the concatenated MSB bits of each pixel. This sequence is divided by CRC-32. The block is not corrupted if the remainder equals zero, else it is corrupted.

Figure 8 presents *Lena* watermarked images of different sizes (128×128 and 256×256, 512×512).

From Figure 8, it is clear that the proposed method based block decomposition generates the aliasing effects and PSNR is very less than PSNR obtained by our scheme based on VD decomposition.

128×128
PSNR=32.00%

256×256
$PSNR = 33.25\%$

512×512
$PSNR = 35.23\%$

Fig. 8. Impact of block decomposition method on the quality of the watermarked image.

5.6 Example of application to medical images

In the literature, most medical image watermarking techniques divide the image manually [7] or automatically [16] into two regions: ROI (Region of Interest) and RONI (Region of Non-Interest). ROI is the part containing the important information to diagnosis. Usually, the information of tamper detection and recovery of ROI are stored in RONI that accepts some visual quality degradation. In some situations, the receiver is powerless to change this partitioning. For example, when he detects ROIs in the RONI. Our proposed scheme can be applicable for medical images and the receiver can specify several ROI and if they have tampered, only tampered packet in these regions are re-transmitted by the sender.

Figure 9 presents an example of medical image[3] decomposed with VD. If we assume that ROIs specified by the receiver are X_8 and X_{17}. Indeed, in the case of tamper, the doctor sent NAK to re-transmit only these regions.

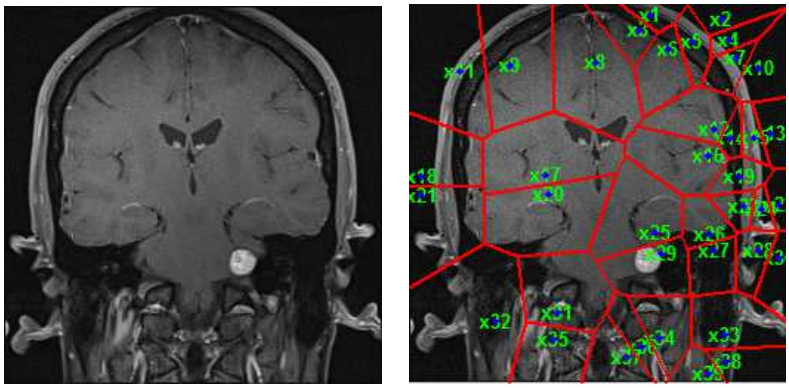
From these results, we can note that the proposed method gives good PSNR and SSIM values, which are interpreted by the good quality of watermarked images. Histograms indicate also the similarity between original and watermarked medical images.

Figure 10 illustrates different scenarios of tampered images and the VM that allows the receiver to detect the tampered regions. Depending on the degree of alteration (number of tampered packets) and the interest of the region, the receiver sends back a negative acknowledgment (NAK) to the sender, requesting that the region should be re-transmitted

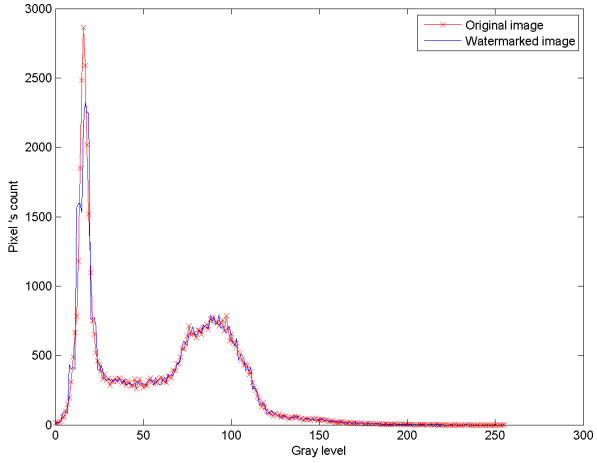
6 Conclusion

In this paper, we propose a novel feature based-fragile watermarking for tamper detection using Voronoi diagram (VD). The Harris Corner detector is used in order

[3] Image download from the free medical image database MedPixhttps://medpix.nlm.nih.gov/



(a) (b)



(c)

Fig. 9. Example of application to medical image: (a) Original medical image. (b) Medical image decomposed using VD. (c) Histograms of the original and watermarked images PSNR = 47.22, SSIM = 0.9851.

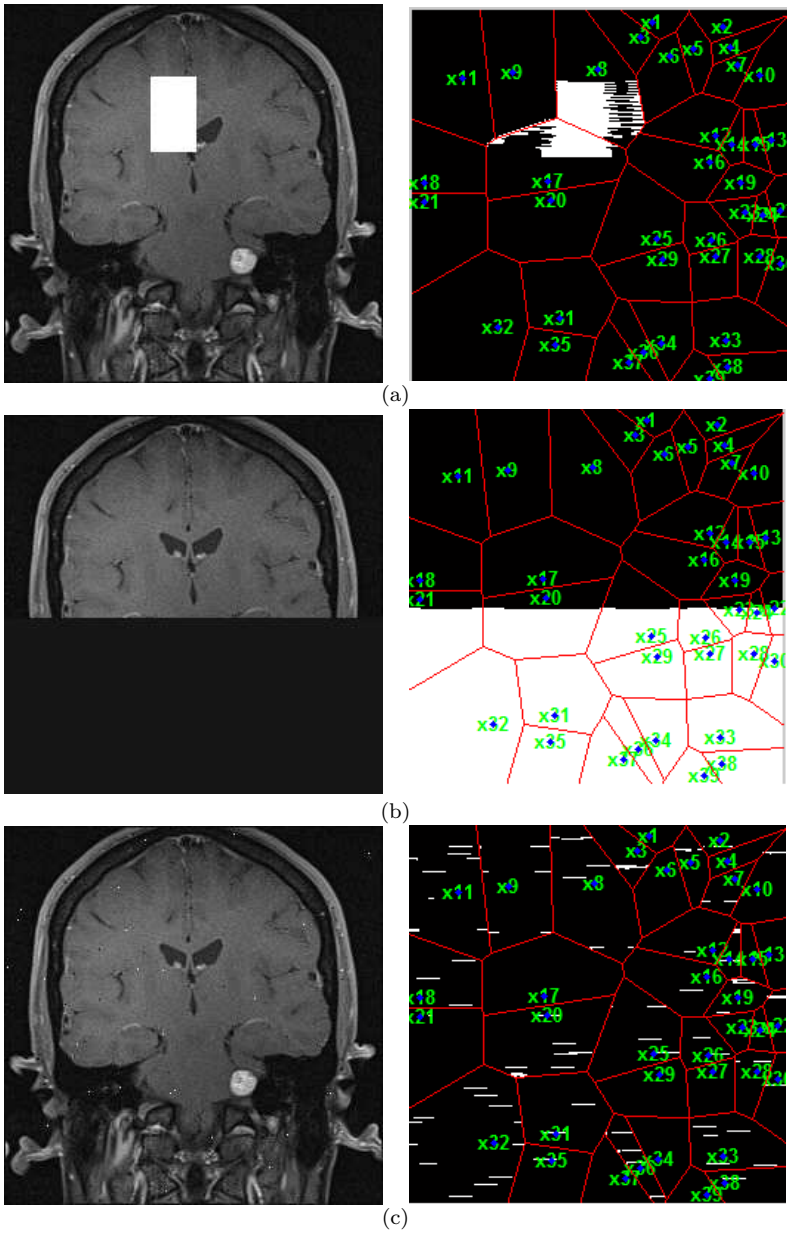


Fig. 10. Different scenarios of tampered medical image: Scenario (a): ROI X_8 and X_{17} have tampered. Scenario (b): ROIs have not tampered. Scenario (c): six packets are tampered in X_8 , no tampered packets in X_{17}.

to extract features points (FP) considered as germs to create Voronoi decomposition of the image. The proposed method is secured using the watermarking public key RSA to encrypt the FP. Our scheme is validated in terms of robustness and efficiency according to the well-known proprieties. Indeed, we have proved that our scheme ensures a watermarking imperceptibility and the fragility criterion. Besides, the proposed watermarking technique is able to locate the tampered regions. Moreover, we have verified that if an unauthorized modification occurred, the location of the corrupted region will be accurately identified using our scheme. In addition to these assessments, we have also evaluated the run time of the proposed scheme comparing with a similar fragile watermarking scheme. We have proved that the VD decomposition is efficiently applied in the proposed scheme in term of the quality of the watermarked image.The proposed method can be applicable in the case where the tamper detection is critical and only some regions of interest must be re-transmitted if they are corrupted, like in the case of medical images.

References

1. Barni, M., Cox, I., Kalker, T., Kim, H.: Digital Watermarking. In: International Workshop, IWDW, Siena, Italy. vol. 3710, pp. 15–17. Lecture Notes in Computer Science (2005)
2. Bas, P., Chassery, J., Macq, B.: Geometrically invariant watermarking using feature points. IEEE Transactions on Image Processing **11**(9), 1014–1028 (2002)
3. Bashir, T., Usman, I., Albesher, A.A., Atawneh, S.H., Naqvi, S.S.: A dct domain smart vicinity reliant fragile watermarking technique for dibr 3d-tv. Automatika **61**(1), 58–65 (2020)
4. Bhattacharjee, S., Kutter, M.: Compression tolerant image authentication. In: International conference image processing. vol. 1, pp. 435–439. IEEE (1998)
5. Chauhan, D., Singh, A.K., Adarsh, A., Kumar, B., Saini, J.: Combining mexican hat wavelet and spread spectrum for adaptive watermarking and its statistical detection using medical images. Multimedia Tools and Applications pp. 1–15 (2017)
6. Durgesh, S., Shivendra, S., Suneeta, A.: Self-embedding pixel wise fragile watermarking scheme for image authentication. In: IITM 2013, CCIS 276. pp. 111–122. Springer-Verlag Berlin Heidelberg (2013)
7. Eswaraiah, R., Reddy, E.S.: Medical image watermarking technique for accurate tamper detection in roi and exact recovery of roi. International journal of telemedicine and applications **2014**, 13 (2014)
8. Golea, N.: A fragile watermarking scheme based CRC checksum and public key cryptosystem for RGB color image authentication. In: 5th International Conference on Image and Signal Processing (ICISP 2012). vol. 7340, pp. 316–325. LNCS, Springer (2012)
9. Goléa, N.E.H., Melkemi, K.E.: Roi-based fragile watermarking for medical image tamper detection. International Journal of High Performance Computing and Networking **13**(2), 199–210 (2019)
10. Hung, K.L., Yen, C.Y.: Watermarking technique based on harris-laplace feature point detector capable of resisting geometric attacks. In: 2019 14th Asia Joint Conference on Information Security (AsiaJCIS). pp. 119–126. IEEE (2019)
11. Huo, Y., Liu, J., Zhang, Q., Zeng, Y., Yang, F., Chang, J., Fan, Y., Liu, C.: Semi-fragile watermarking for color image authentication in power internet of things. In: 2019 IEEE Innovative Smart Grid Technologies-Asia (ISGT Asia). pp. 2865–2869. IEEE (2019)
12. Kougianos, E., Mohanty, S.P., Mahapatra, R.N.: Hardware assisted watermarking for multimedia. Computers and Electrical Engineering **35**(2), 339–358 (2009)

13. Kutter, M., Bhattacharjee, S., Ebrahimi, T.: Towards second generation watermarking schemes. In: International conference image processing. vol. 1, pp. 320–323. IEEE (1999)
14. Lee, H., Kim, H., Lee, H.: Robust image watermarking using local invariant features. Optical Engineering - The Journal of SPIE (The International Society for Optical Engineering), U.S.A. **45**(3), 1–11 (2006)
15. Lin, P.L., Huang, P.W., Peng, A.W.: A fragile watermarking scheme for image authentication with localization and recovery. In: IEEE Sixth International Symposium on Multimedia Software Engineering. pp. 146–153. IEEE (2004)
16. Memon, N.A., Chaudhry, A., Ahmad, M., Keerio, Z.A.: Hybrid watermarking of medical images for roi authentication and recovery. International Journal of Computer Mathematics **88**(10), 2057–2071 (2011)
17. Niu, P.p., Wang, L., Shen, X., Zhang, S.y., Wang, X.y.: A novel robust image watermarking in quaternion wavelet domain based on superpixel segmentation. Multidimensional Systems and Signal Processing pp. 1–22 (2020)
18. Sahu, N., Sur, A.: Sift based video watermarking resistant to temporal scaling. Journal of Visual Communication and Image Representation **45**, 77–86 (2017)
19. Seo, J., Yoo, C.: Localized image watermarking based on feature points of scale-space representation. Pattern Recognition **37**(7), 1365–1375 (2004)
20. Su, K., Kundur, D., Hatzinakos, D.: Spatially localized image dependent watermarking for statistical invisibility and collusion resistance. IEEE Transaction on Multimedia **7**(1), 52–66 (2005)
21. Su, Q., Chen, B.: Robust color image watermarking technique in the spatial domain. Soft Computing **22**(1), 91–106 (2018)
22. Suhail, M., Obaidat, M.: Digital Watermarking-Based DCT and JPEG Model. IEEE Transactions on Instrumentation and Measurement **52**(5), 1640–1647 (2003)
23. Tang, C., Hang, H.: A feature-based robust digital image watermarking scheme. IEEE Transactions on Image Processing **51**(4), 950–959 (2003)
24. Tommasini, T., Fusiello, A., Trucco, E., Roberto, V.: Marking good features track better. In: Computer Vision and Pattern Recognition. pp. 178–183. IEEE (1998)
25. Wang, X., Hou, L., Wu, J.: A feature-based robust digital image watermarking against geometric attacks. Image and Vision Computing **26**(7), 980–989 (2008)
26. Wang, X., Yang, Y., Yang, H.: Invariant image watermarking using multiscale harris detector and wavelet moments. Comput. Electr. Eng **36**(41), 31–44 (2010)
27. Wei Lu, H.L., Chung, F.L.: Feature based robust watermarking using image normalization. Computers & Electrical Engineering **36**(1), 2–18 (2010)
28. Xiaojun, Q., Ji, Q.: A Robust Content-Based Digital Image Watermarking Scheme. Signal Processing **87**(6), 1264–1280 (2007)
29. Yuan, X.C., Pun, C.M.: Geometrically invariant image watermarking based on feature extraction and zernike transform. International Journal of Security and Its Applications **6**(2), 217–222 (2012)