Self-Embedding Watermarking Scheme Based on MDS Codes

Dongmei Niu[1,2(✉)], Hongxia Wang[1], Minquan Cheng[1], and Linna Zhou[3]

[1] Southwest Jiaotong University, Chengdu, 610031, China
niudongmei2007@163.com, hxwang@home.swjtu.edu.cn
[2] Southwest University of Science and Technology, Mianyang 621010, China
[3] University of International Relations, Beijing 100091, China

Abstract. This paper proposes a self-embedding watermarking scheme based on Maximum Distance Separable (MDS) codes. The watermark is comprised of the reference-bits and the authentication-bits. The reference-bits is generated by encoding the principal content of all the image blocks. The encoding matrix is derived from the generator matrix of selected systematic MDS code. Based on this encoding method, the reference-bits embedded in an image block will be shared by all the image blocks. Therefore our scheme realize a new reference share mechanism and is immune to the tampering coincidence and the reference waste. Moreover, the maximal tampering rate can be analyzed from the error resilience of the MDS code. On the receiver side, the tampered image blocks can be located by the embedded authentication-bits. As long as the tampering rate is not larger than the maximal tampering rate, the principal content of the tampered image blocks can be recovered perfectly. The restoration is deterministic and the quality of recovered content is constant. Our experimental results demonstrate that the proposed method outperforms the recently state-of-the-art works.

Keywords: Self-embedding · Image authentication · Fragile watermarking · MDS codes · Cauchy matrix

1 Introduction

Self-embedding watermarking scheme has been proposed in [1] for detecting the tampered image regions and recovering the tampered content. In most self-embedding watermarking schemes, the original image will be divided into blocks. In addition to the authentication-bits for detecting the tampered image blocks, the reference-bits for recovering the tampered image blocks is embedded in the image [2]. The reference-bits is usually the representative information of the host image blocks such as the prime DCT coefficients, the MSB of all pixels in the image block, and the vector quantization values. In some schemes, for example in [1, 3–6], the reference-bits of an image block is usually embedded into another different image block. Usually a block-mapping is needed to determine the embedding position. This method will inevitably lead to the problems of tampering coincidence and the reference waste [7]. In some schemes [8–11], the reference-bits will be duplicated and embedding in the image for many times to reduce

Y.-Q. Shi et al. (Eds.): IWDW 2015, LNCS 9569, pp. 250–258, 2016.
DOI: 10.1007/978-3-319-31960-5_21

the probability of the tampering coincidence, while the cost of reference waste will increase accordingly.

In [12, 13], a reference-sharing mechanism is proposed to avoid the tampering coincidence and the reference waste problems. In the schemes, the reference-bits embedded in an image block is generated by encoding the principal content in different blocks and shared by these blocks for content restoration, which can achieve good recovery performance even higher tampered rate. This thought is also reflected in the other schemes [14–16]. In [17], the content reconstruction problem is modeled as a communication over an erasure channel. The reference information blocks are generated by encoding the reference symbols blocks of all the image blocks based on Random Linear Fountain (RLF) codes. So, the reference information block embedded in an image block will be shared by all the image blocks, which allows for working with higher tampering rates than other self-embedding schemes with the same rate of reference information per image block. To resolve the problems of the tampering coincidence and reference waste, both [13, 17] adopted the reference-sharing method based on different spreading mechanism. However, the tampered image blocks can only be recovered perfectly with a great probability by using the methods in [13, 17]. In this paper, we propose a deterministic self-embedding watermarking scheme based on MDS codes. As long as the tampering rate is not larger than the maximal tampering rate, the restoration will be perfect absolutely.

2 Watermark Embedding Procedure

Similar to the common self-embedding schemes, the watermark data of the proposed scheme is made up of two parts: the reference-bits and the authentication-bits. In the watermark embedding procedure, we first select a suitable MDS code, then encode the 5 most significant bits (MSB) of all pixels in the image blocks by using the MDS code to generate the reference-bits. The authentication-bits is the hash-bits determined by both the MSB of the image block and the reference-bits. The reference-bits and the hash-bits will replace the 3 least significant bits (LSB) of all pixels in the image block.

2.1 Reference-Bits Generation

Assume the original image is divided into blocks sized 8×8 pixels. The number of the blocks is denoted as K. For each image block, we collect the 5 MSB of all pixels in the block to form a column vector. There will be K vectors in total. We denote them as $(D_1, D_2, \ldots, D_K)$. The length of each vector is 320. The reference-bits vectors will be generated by encoding the vectors based on MDS code and embedded as part of watermark into the 3LSB planes of the image block. We use 160 bits to store the reference-bits. So, the length of the reference-bits vector is 160. The ratio of the length of the reference-bits vector to the length of MSB vector is denoted as R. The value of R will determine which MDS code will be used.

Here $R = 1/2$, we will calculate the reference-bits vectors based on the systematic $(3\ K,\ 2\ K)$-MDS code over the finite field. First, we divide $D_i (i = 1, 2, \ldots, K)$ into 2

shorter vectors $\boldsymbol{D}_{i1}, \boldsymbol{D}_{i2}$. There will be 2 K shorter vectors in total. Then, we encode the 2 K shorter vectors base on the (3 K, 2 K)-MDS code in the following way:

$$\left(C_1, C_2, \ldots, C_K\right) = \left(D_{11}, D_{12}, D_{21}, D_{22}, \ldots, D_{K1}, D_{K2}\right) A_{2K\times K}, \tag{1}$$

where $\boldsymbol{A}$ is the 2 K rows and K columns matrix and ($\boldsymbol{I}|\boldsymbol{A}$) is the generator matrix of the systematic (3 K, 2 K)-MDS code over the finite field. The calculation will be done over the finite field. For this purpose, $D_{ij}(i = 1, \ldots, K, j = 1, 2)$ will be transformed to an n-dimensional column vector in the finite field. For example, $\boldsymbol{D}_{11}$ is transformed to $(d_{11},d_{21},\ldots,d_{n1})^T$. So, we can rewrite (1) as,

$$\left(C_1, C_2, \ldots, C_K\right) = \begin{bmatrix} d_{11} & d_{12} & \ldots & d_{1,2K} \\ d_{21} & d_{22} & \ldots & d_{2,2K} \\ \vdots & \vdots & \cdots & \vdots \\ d_{n1} & d_{n2} & \ldots & d_{n,2K} \end{bmatrix} A_{2K\times K}. \tag{2}$$

From (2), we can see that $\boldsymbol{C}_i$ ($i = 1,\ldots,K$) is an n-dimensional column vector in the finite field. Finally, we transform $\boldsymbol{C}_i$ ($i = 1,\ldots,K$) to binary vector. The transformed binary vectors are denoted as ($\boldsymbol{R}_1,\boldsymbol{R}_2,\ldots,\boldsymbol{R}_K$), which are the reference-bits vectors.

From (1) it can be seen that $\boldsymbol{C}_i$ ($i = 1,\ldots,K$) is the linear combination of all the MSB vectors. That means $\boldsymbol{C}_i$ ($i = 1,\ldots,K$) or the reference-bits vector $\boldsymbol{R}_i$ ($i = 1,\ldots,K$) carries the information of all the image blocks. The reference-bits vector $\boldsymbol{R}_i$ will be shared as the recovery information by all the image blocks. So, a new reference share mechanism is realized based on the MDS codes.

2.2 Authentication-Bits Generation

For the ith ($i = 1,\ldots,K$) image block, the MSB vector $\boldsymbol{D}_i$ and the reference-bits $\boldsymbol{R}_i$ are connected and then fed into a hash function to generate the 32 hash bits vector $\boldsymbol{H}_i$. The vectors $\{\boldsymbol{H}_1,\boldsymbol{H}_2,\ldots,\boldsymbol{H}_K\}$ is the authentication-bits which will be embedded into the 3LSB of all pixels in the image block as a part of the watermark. In our experiment, we use the MD5 function, the output is shortened by exclusive disjunction on neighboring bit pairs to generate the required length hash bits.

2.3 Watermark-Bits Embedding

For the ith ($i = 1,\ldots,K$) image block, the 160 reference-bits $\boldsymbol{R}_i$ and the 32 authentication-bits H_i are connected and permuted based upon the secret key to generate the 192 watermark bits $\boldsymbol{W}_i$, which will be used to replace the 3LSB of all pixels in the ith image block. After all the image blocks have been processed, the watermarked image is produced. The entire procedure of watermark embedding can be sketched in the Fig. 1.

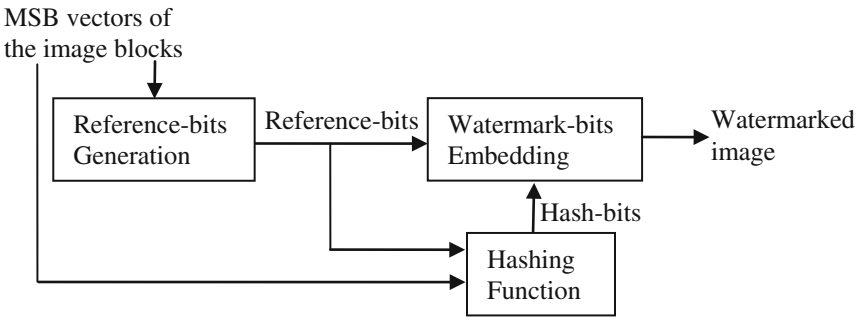

Fig. 1. The procedure of watermark embedding

3 Tampering Detection and Content Recovery Procedures

On the receiver side, the received image will be divided into blocks with the same size as the original image. One can identify if the image block is tampered or not and locate the tampered image blocks by the authentication data. The ratio between the number of tampered image blocks and the number of all blocks is called as the tampering rate, the maximal tampering rate is the upper bound of the tampering rate. As long as the tampering rate is not larger than the maximal tampering rate, the representative data of the tampered image blocks can be recovered perfectly. In our work, the maximal tampering rate can be easily derived based on the superior error-correcting characteristics of the MDS codes, which will be discussed in detail in Sect. 3.3.

3.1 Tampered Blocks Detection

For the ith image block, the watermark bits is extracted from the redundant space, scrambled inversely using the same secret key and decomposed into two parts: the reference-bits vector $\boldsymbol{R}_i$ and the hash-bits vector $\boldsymbol{H}_i$. If the recalculated hash value of the 5MSB vector $\boldsymbol{D}_i$ and the extracted reference-bits differs from the extracted hash value, the ith image block is judged to be "tampered", that is, some content in the image block has been modified. Otherwise, we say it is a "reserved" [13]. As long as the tampering rate is not larger than the maximal tampering rate, we can perfectly recover the failed 5MSB of the tampered image blocks by the decoding method of the systematic MDS codes. The decoding procedure can be illustrated as follows.

3.2 Content Recovery

After identifying the tampered image blocks, we extract the reference-bits vectors blocks from the reserved image blocks. Suppose the number of reserved image blocks is r. So, we can extract r reference-bits vectors, which are denoted as $(\boldsymbol{C}_{e(1)},\boldsymbol{C}_{e(2)},\ldots,\boldsymbol{C}_{e(r)})$. Then we can rewrite (1) as,

$$\left(C_{e(1)}, C_{e(2)}, \ldots, C_{e(r)}\right) = \left(D_{11}, D_{12}, D_{21}, D_{22}, D_{2,2}, \ldots, D_{K1}, D_{K2}\right) A^{(E)}_{2K\times K}, \tag{3}$$

where $A^{(E)}_{2K\times K}$ is the matrix with columns taken from $\boldsymbol{A}_{2K\times K}$ corresponding to extractable reference-bits vectors. Note that 5MSB vectors of the reserved image blocks which can be obtained, while 5MSB vectors of the tampered image blocks are unknown. By denoting the 5MSB vectors of the tampered and reserved blocks as $\boldsymbol{D}_T$ and $\boldsymbol{D}_R$, respectively, we can reformulate (3) as follows,

$$\left(C_{e(1)}, C_{e(2)}, \ldots, C_{e(r)}\right) - D_R A^{(E,R)}_{2K\times K} = D_T A^{(E,T)}_{2K\times K}, \tag{4}$$

where $A^{(E,R)}_{2K\times K}$ and $A^{(E,T)}_{2K\times K}$ are matrices whose rows are those in $A^{(E)}_{2K\times K}$ corresponding to the 5MSB vectors in $\boldsymbol{D}_R$ and $\boldsymbol{D}_T$, respectively. In (4), the left side and matrix $A^{(E,T)}_{2K\times K}$ are known, and our purpose is to find the $\boldsymbol{D}_T$. Denote the length of $\boldsymbol{D}_T$ as n_T so that the size of $A^{(E,T)}_{2K\times K}$ is $n_T \times r$. We will solve the n_T unknowns according to the r equations over the finite field. Actually, it can be demonstrated that if the tampering rate is not larger than the maximal tampering rate, there will be $n_T \le r$. This implies that the number of equations is more than the number of the unknowns. We can rewrite (4) as,

$$\left(C_{e(1)}, C_{e(2)}, \ldots, C_{e(n_T)}\right) - D_R A^{(E,R,n_T)}_{2K\times K} = D_T A^{(E,T,n_T)}_{2K\times K} \tag{5}$$

where $A^{(E,T,n_T)}_{2K\times K}$ is the $n_T \times n_T$ matrix whose columns are the first n_T columns of the matrix $A^{(E,T)}_{2K\times K}$, $A^{(E,R,n_T)}_{2K\times K}$ is the first n_T columns of the matrix $A^{(E,R)}_{2K\times K}$. In the left side of (5), $(\boldsymbol{C}_{e(1)},\boldsymbol{C}_{e(2)},\ldots,\boldsymbol{C}_{e(n_T)})$ is the first n_T data block of $(C_{e(1)},C_{e(2)},\ldots,C_{e(r)})$. It is noticeable that the matrix $A^{(E,T,n_T)}_{2K\times K}$ is the square submatrix of $\boldsymbol{A}$. So, $A^{(E,T,n_T)}_{2K\times K}$ will be nonsingular because $(\boldsymbol{I}|\boldsymbol{A})$ is the generator matrix of systematic MDS. Therefore, the Eq. (5) has an unique solution. We can solve the Eq. (5) over the finite field to retrieve the original values of $\boldsymbol{D}_T$. That implies we can retrieve the original values of $(\boldsymbol{D}_{11},\boldsymbol{D}_{12}, \boldsymbol{D}_{21},\boldsymbol{D}_{22},\ldots, \boldsymbol{D}_{K1},\boldsymbol{D}_{K2})$. So, we can recover the K MSB vectors $(\boldsymbol{D}_1, \boldsymbol{D}_2,\ldots,\boldsymbol{D}_K)$.

The recovered MSB vectors can be used to reconstruct the tampered image blocks. Provided that the tampering rate is not larger than the maximal tampering rate, the quality of the reconstructed image areas will be constant. That is the quality of the reconstructed content does not degrade with the tampering area increasing.

3.3 Analysis of the Maximal Tampering Rate

From the previous analysis, we generate the K reference-bits vectors by encoding the 2 K shorter vectors based on the systematic (3 K, 2 K)-MDS code. We know that the (3 K, 2 K)-MDS code is capable of being resilient to arbitrary K failures. That is, the system can recover arbitrary K failures happening in the K reference-bits vectors and the 2 K shorter vectors. But it need to be noted that if an image block is identified as a tampered block, there will be 2 shorter vectors and 1 reference-bits vector is identified as the failed data

blocks. This means there will be 3 failures happening. So, the proposed scheme can only recover arbitrary $K/3$ image blocks failures. There are K image blocks in total. Therefore, the maximal tampering rate of our scheme is 1/3.

4 Experimental Results and Comparisons

8-bit gray scale image Lake sized 512×512 is used as the host. The number of image blocks $K = 2^{12}$. To produce the reference data blocks we need a systematic $(3 \times 2^{12}, 2 \times 2^{12})$-MDS code. Assume its generator matrix is $(\boldsymbol{I}|\boldsymbol{A})$, $\boldsymbol{A}$ is an $2^{13} \times 2^{12}$ matrix. Here we generate the matrix $\boldsymbol{A}$ by constructing the $2^{13} \times 2^{12}$ Cauchy matrix over $G(2^{16})$. The 5MSB vector of each image block is divided into two short vectors size of 160 bits. So there are 2^{13} shorter vectors in total. Each vector will be represented as a column vector of 10 elements in the finite field $G(2^{16})$. Then we calculate the 2^{12} reference-bits vectors according to (2). Each reference data block will be a column vector of 10 elements in the finite field $G(2^{16})$ and can be transformed into a binary vector of length 160 bits.

(a) (b)

Fig. 2. (a) Original image Lake. (b) Watermarked image Lake produced by $R = 1/2$.

Figure 2(a) gives the Lake image. Figure 2(b) give the watermarked Lake in the experiment. The values of PSNR due to watermark embedding are 37.9 dB. Figure 3 shows three tampered versions of watermarked Lake with different tampering rates, and their corresponding identification and restoration results in the first experiments. We can see when the tampering rate $\alpha = 9.8$ %, 21.83 % and 32.69 %, all tampered blocks are located correctly. The tampered blocks are represented by the extreme white. The original MSB of tampered blocks are recovered without any error. In the three cases, PSNR values in the restored area are all 40.7 dB when regarding original image as reference. The quality of the recovered content does not degrade with the growth of tampering rate. Here, just like the method used in [13], forcing the first and second LSB as 0 and the third LSB as 1. The experiment demonstrate than if the ratio $R = 1/2$, the proposed scheme can perfectly recover the representative data of the tampered image blocks as long as the tampering rate is not larger than 1/3.

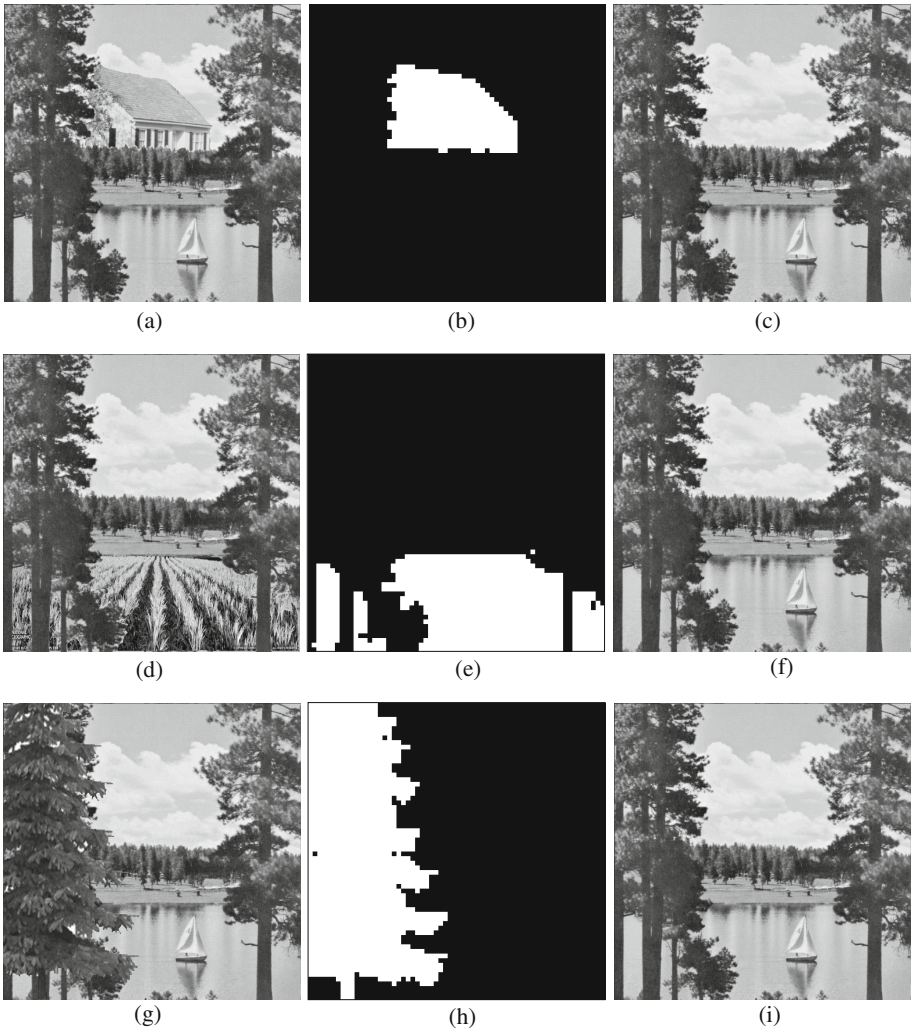

Fig. 3. (a) Tampered Lake with $\alpha = 9.8$ %. (b) Tampered blocks identification result of (a). (c) Restored version of (a). (d) Tampered Lake with $\alpha = 21.83$ %. (e) Tampered blocks identification result of (d). (f) Restored version of (d). (g) Tampered Lake with $\alpha = 32.69$ %. (h) Tampered blocks identification result of (g). (i) Restored version of (g).

Finally, we compares the restoration capability of the proposed scheme in the experiment with the methods in [13, 17]. The experimental parameters of the three methods is the same. The performance comparison is made by three main evaluation indexes: the watermarked image quality, the maximal tampering rate and the restored image quality. All the three methods exploit 3 LSB watermark embedding. Therefore, the PSNR due to watermarking embedding is identical and equals 37.9 dB. The reference-bits vectors are all 160 bits and generated by encoding the 5MSB of all pixels in the 8×8 image

blocks. When the tampering rate is not larger than the maximal tampering rate, all the three methods can recover the 5MSB vectors of the tampered image blocks. PSNR values in restored area is identical and equals 40.7 dB when regarding original image as reference. But the maximal tampering rate of our proposed method is 33 % which is better than 24 %, the maximal tampering rate of the method in [13] and equals to that of the method in [17]. However, the biggest advantage of the proposed method is our encoding matrix is derived from the generator matrix of the systematic MDS code. The property of the MDS code can promise the restoration can be successful absolutely, but the encoding matrix applied in the methods in [13, 17] are random matrix. The random matrix can only promise the restoration could be successful with a great probability. So, the proposed method offers a deterministic self-embedding scheme which has the same performance comparing to the method in [17], while increasing the maximal tampering rate comparing to the method in [13].

5 Conclusions

This paper proposed a self-embedding watermarking scheme based on MDS codes. The scheme realizes a new reference sharing mechanism to resist the tampering coincidence and the reference waste. Based on our model, the maximal tampering rate can be derived from the error resilience of MDS code. As long as the tampering rate is not larger than the maximal tampering rate, the representative data of the tampered image blocks can be recovered absolutely. The quality of the recovered content is constant. Our theoretical analysis and experimental results demonstrate that the proposed method outperforms the recently state-of-the-art works.

Acknowledgements. This work is supported in part by the National Natural Science Foundation of China (NSFC) (Nos. 61170226, 61170175), and is supported in part by Guangxi Natural Science Foundation under Grant No. 2013GXNSFCA019001.

References

1. Fridrich, J., Goljan, M.: Images with self-correcting capabilities. In: Proceeding of IEEE International Conference on Image Processing, pp. 792–796 (1999)
2. Korus, P., Dziech, A.: Adaptive self-embedding scheme with controlled reconstruction performance. IEEE Trans. Inf. Forensics Secur. **9**(2), 1134–1147 (2014)
3. Yang, C.W., Shen, J.J.: Recover the tampered image based on VQ indexing. Signal Process. **90**(1), 331–343 (2010)
4. Huo, Y., He, H., Chen, F.: Alterable capacity fragile watermarking scheme with restoration capability. Opt. Commun. **285**(7), 1759–1766 (2012)
5. Qin, C., Chang, C.-C., Chen, P.-Y.: Self-embedding fragile water-marking with restoration capability based on adaptive bit allocation mechanism. Signal Process. **92**(4), 1137–1150 (2012)
6. He, H., Chen, F., Tai, H.M., Kalker, T., Zhang, J.: Performance analysis of a block-neighborhood-based self-recovery fragile watermarking scheme. IEEE Trans. Inf. Forensics Secur. **7**(1), 185–196 (2012)

7. Zhang, X., Qian, Z., Ren, Y., Feng, G.: Watermarking with flexible self-recovery quality based on compressive sensing and compositive reconstruction. IEEE Trans. Inf. Forensics Secur. **6**(4), 1223–1232 (2011)
8. Lee, T.Y., Lin, S.: Dual watermark for image tampering detection and recovery. Pattern Recogn. **41**(11), 3497–3506 (2008)
9. Li, C., Wang, Y., Ma, B., Zhang, Z.: A novel self-recovery fragile watermarking scheme based on dual-redundant-ring structure. Comput. Electr. Eng. **37**(6), 927–940 (2011)
10. Qin, C., Chang, C.C., Hsu, T.J.: Effective fragile watermarking for image authentication with high-quality recovery capability. KSII Trans. Internet Inf. Syst. **7**(11), 2941–2956 (2013)
11. Qin, C., Chang, C.C., Chen, K.N.: Adaptive self-recovery for tampered images based on VQ indexing and inpainting. Signal Process. **93**(4), 933–946 (2013)
12. Zhang, X., Wang, S., Feng, S.G.: Fragile watermarking scheme with extensive content restoration capability. In: Proceeding of International Workshop Digital Watermark, pp. 268–278 (2009)
13. Zhang, X., Wang, S., Qian, Z., Feng, G.: Reference sharing mechanism for watermark self-embedding. IEEE Trans. Image Process. **20**(2), 485–495 (2011)
14. Zhang, X., Wang, S.: Fragile watermarking with error free restoration capability. IEEE Trans. Multimedia **10**(8), 1490–1499 (2008)
15. Zhang, X., Wang, S.: Fragile watermarking scheme using a hierarchical mechanism. Signal Process. **89**(4), 675–679 (2009)
16. Qian, Z., Feng, G., Zhang, X., Wang, S.: Image self-embedding with high-quality restoration capability. Digit. Signal Process. **21**(2), 278–286 (2011)
17. Korus, P., Dziech, A.: Efficient method for content reconstruction with self-embedding. IEEE Trans. Image Process. **22**(3), 1134–1147 (2013)
18. MacWilliams, F.J., Sloane, N.J.A.: The Theory of Error Correcting Codes. North-Holland, Amsterdam (1977)