 CMC, vol.57, no.2, pp.243-260, 2018

Self-embedding Image Watermarking based on Combined Decision Using Pre-offset and Post-offset Blocks

Daofu Gong[1, 2], Yan Chen[1, 2], Haoyu Lu[1, *], Zhenyu Li[3] and Yibing Han[1]

Abstract: To detect and recover random tampering areas, a combined-decision-based self-embedding watermarking scheme is proposed herein. In this scheme, the image is first partitioned into 2×2 size blocks. Next, the high 5 bits of a block's average value is embedded into its offset block. The tampering type of block is detected by comparing the watermarks of its pre-offset and post-offset blocks. The theoretical analysis and experiments demonstrate that the proposed scheme not only has a lower ratio of false detection but also better performance with regard to avoiding random tampering.

Keywords: Fragile watermarking, self-embedding, offset block, tamper recovery, random tampering.

1 Introduction

Digital images, one of the main carriers for obtaining and disseminating information, are of great convenience to mankind. However, their feature of being easily edited and modified introduces several security risks. Therefore, authenticity verification, locating tampered areas, and tamper recovery, all constitute important research branches of information security.

Self-embedding watermarking as a branch of information hiding is a technique of encoding the image itself as watermark information. It evaluates the change in the watermark extracted from the tampered image to verify its integrity and recovers the tampered image area using the self-encoding image information. The block-based method is one of the research focuses in self-embedding image authentication technology. The method divides the image into blocks, and for each block, it generates an authentication watermark and a restoration watermark to locate the tampering and restore the image at the block level. Hence, locating accuracy and algorithm security can be balanced by the size of the block.

Lin et al. [Lin, Hsieh and Huang (2005)] proposed a block-based multi-level fragile watermarking algorithm. The algorithm first divides the original image into 4×4 and selects the offset blocks of each block. Then, each block is further divided into 2×2

[1] Zhengzhou Science and Technology Institute, Zhengzhou, 450001, China.
[2] State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou, 450001, China.
[3] Department of Computer Science, University of York, York YO10 5GH, UK.
[*] Corresponding Author: Haoyu Lu. Email: galuowazhiye@gamil.com.

blocks, and the authentication and restoration watermarks are calculated. Finally, the watermarks are embedded in the offset blocks of the 4×4 blocks.

However, He et al. [He and Chen (2007); Chang, Fan and Tai (2008)] pointed out a security risk due to the small mapped key space, and proposes a forgery attack against the method proposed by Lin et al. [Lin, Hsieh and Huang (2005)]. Wang et al. [Wang, Liu, Liu et al. (2008)] proposed a fragile watermarking algorithm, which can examine the tampering and recover the tampered area under certain conditions. First, the image is divided into 8×8 blocks, and the corresponding offset sub-block is determined using chaotic mapping. Then, the recovery watermark is generated from the DCT coefficients of the sub-block and embedded in the sub-lowest bit of the offset block. Finally, the authentication watermark is embedded in the lowest bit of the sub-block.

Duan et al. [Duan, Zhao, Li et al. (2010)] proposed a recoverable fragile watermarking algorithm based on the Slant transform. Based on the independent blocking technique, this algorithm embeds the watermark bits for authentication into the mid-frequency region of the Slant transform domain of each block. The compressed image data is embedded as the restoration watermark in the lowest bit of the pixel. The tamper localization and restoration accuracy are all at 8×8 block level.

In Yang et al. [Yang and Cai (2011)], a fragile watermarking algorithm based on block recovery is proposed, which first divides the image into 2×2 blocks and performs singular value decomposition on the highest 6 bits of each block. Then, after chaos scrambling (as in Wang et al. [Wang, Liu, Liu et al. (2008)]) is performed on the block, it embeds the restoration watermark in the second lowest position of the image block and embeds the authentication watermark in the least significant bit of the image block. The accuracy of the tamper localization and recovery is at 2×2 block level. In Chang et al. [Chang and Tai (2013)], authentication and restoration watermarks were generated for each image block and used for the detection of the image block neighborhood, as well as the hierarchical structure, to detect the tampering of the image block.

Deng et al. [Deng, Chen, Zeng et al. (2013)] proposed a tamper detection and recovery algorithm for medical images. The algorithm divides the image layer by layer using a quadtree, then generates restoration and certification watermarks for each block and embeds them in the LSB of the image. In Chen et al. [Chen, He, Huo et al. (2011)] a variable-capacity self-recovery watermarking algorithm is proposed. The watermark of the algorithm consists of a 24-bit basic watermark and a variable-length restoration watermark. In Kiatpapan et al. [Kiatpapan and Kondo (2015)], a dual watermark algorithm is proposed for the authentication and restoration of color images. In Dhole et al. [Dhole and Patil (2015)], a self-embedding watermarking algorithm based on blockchain is proposed for authentication and recovery of 8×8 image blocks. In Singh et al. [Singh and Singh (2016)], a self-recovery fragile watermark based on the DCT coefficients of a 2×2 image block is proposed. It generates a 2-bit authentication watermark and a 10-bit restoration watermark for each image block, which should be embedded in the lowest 3 bits of the offset blocks. For this reason, such watermark embedding significantly affects image quality. Qin et al. [Qin, Ji, Wang et al. (2017)] proposed a self-recovery fragile watermark based on vector quantization. Using vector quantization to generate image content information and reference watermark information, this method implements authentication and recovery of an 8×8 image block. In Cao et al. [Cao, An, Wang et al. (2017)], a sub-embedded watermarking algorithm based on hierarchical recovery is proposed; it determines which content information to be restored first according to the importance of the image MSBs.

The self-embedding fragile watermarking algorithms mentioned above generate an authentication watermark and a restoration watermark for each block of the image, while He et al. [He, Zhang and Chen (2008); Chen, He and Wang (2012); He, Chen, Tai et al. (2012)] use the image content feature as a watermark for both authentication and recovery, causing the length of the watermark to be reduced. He et al. [He, Zhang and Chen (2008); Chen, He and Wang (2012)] proposed a self-embedding watermarking algorithm based on the Chinese Remainder Theorem. In this algorithm, a DCT transform is first performed on each 8×8 block, and DCT coefficients are quantized to generate restoration watermark information, which is embedded in the lower bits of the offset blocks. Then the authentication is performed by comparing the mean error between the reconstructed image blocks and the compressed image blocks. However, the algorithm's authentication and recovery accuracy is only at the 8×8 block level and has a higher misjudgment rate. In He et al. [He, Zhang and Chen (2008)], the content information of each 2×2 image block is quantized into 5 bits as a watermark. Tamper detection is performed by comparing the match/matching degree of eight neighborhood characteristics of each block and the watermark. This paper has improved the methods described in Chen et al. [Chen, He and Wang (2012); He, Chen, Tai et al. (2012)]. In Chen et al. [Chen, He and Wang (2012)], a variable-capacity self-embedding watermarking algorithm is proposed, which enables resistance against constant mean attacks. In He et al. [He, Chen, Tai et al. (2012)], a variety of optimizations has been carried out on tamper detection, boosting the performance of the algorithm for area tampering. However, the algorithms in He et al. [He, Zhang and Chen (2008); Chen, He and Wang (2012); He, Chen, Tai et al. (2012)] have a high false negative rate and poor performance in detecting random tampered images.

To solve the problem of detection and recovery of random tampering, as well as the credibility of the recovery, this paper proposes a self-embedding fragile watermarking algorithm based on the combined decision of pre-offset and post-offset blocks. The algorithm utilizes the matching of the feature watermark and extracted watermark of its pre-offset and post-offset blocks to determine tampering. It judges if the post-offset block is tampered with to determine whether it can be recovered. Theoretical analysis and experiments show that the proposed algorithm effectively reduces the false detection rate and can accurately locate and recover the random tampering of the image.

2 The proposal of the idea

In the neighborhood-based self-recovery watermarking method proposed in He et al. [He, Zhang and Chen (2008); Chen, He and Wang (2012); He, Chen, Tai et al. (2012)], the image is first divided into 2×2 blocks, and the content information of the block is quantized by 5 bits and embedded as watermark information into its offset sub-block. The basic principle of judging if an image block has been tampered with is shown in Fig. 1, assuming that the watermark information of 16 image blocks in area A is randomly embedded in 16 gray image blocks (noted as set B). Correspondingly, the watermark information of the 16 black image blocks (noted as set C) is embedded in area A. Since the watermark information corresponding to the image blocks in set C will be changed once area A is tampered with, the image blocks in area A and set C will be detected rather than set B. Therefore, tampered area A must be distinguished from the un-tampered set C. As can be seen from Fig. 1, the number of tampered image blocks around the image block in area A is greater than the number of tampered image blocks around the image block in set C. Then by comparing the number of detected image blocks in the 8-neighborhood of the measured image block and its offset block, the detected image blocks in area A and set C can be distinguished, thereby reducing the probability of false alarms. Afterwards, the watermark information of area A embedded in set B will be used for recovery.

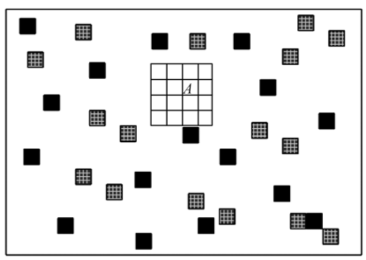


Figure 1: Watermark embedding position diagram in He et al. [He, Chen, Tai et al. (2012)]

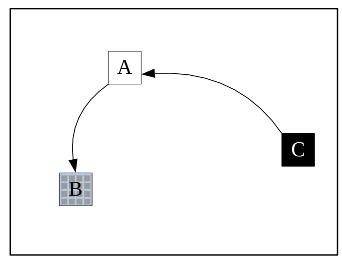


Figure 2: Watermark embedding position diagram in this paper

This idea can effectively detect area tampering, but a single image block being modified (i.e. random tampering), is difficult. For example, if area A is a single image block, the number of detected blocks of image block A and neighboring area C is 0, so that the tampering of A and C cannot be distinguished. At the same time, if the watermark information contained in set B is tampered with while the image block in the area A is still being used to recover, an erroneous, and therefore unreliable recovery, will occur.

To solve this problem, this paper carries out tamper detection and recovery using an offset block combined decision. To illustrate the basic principles of the algorithm, a single image block is adopted as an example. As shown in Fig. 2, the watermark of image block A is embedded in image block B, and the watermark of image block C is embedded in image block A. Image block B is herein referred to as the “pre-offset block” of image block A, and image block C is called the “post-offset block” of image block A. The watermark generated by the higher bits information of the image block is referred to as a “feature watermark”, and the watermark information extracted from the lower bits of the pre-offset block is referred to as “extracted watermark”.

In fact, simply comparing the feature watermark with the extracted watermark will result in set A and set C being indistinguishable. This is because the content of the image block in set A has been tampered, thereby causing its feature watermark to change and as a result, the feature watermark and the extracted watermark cannot be matched. For the image block in set C, the extracted watermark contained in set changes with the tampering of set A, so that the feature watermark and the extracted watermark cannot be matched. Therefore, for situations in which the extracted watermark of the image block and of the content feature being tampered can be distinguished, the distinction of sets A and C can be completed, and the problem of false recovery caused by a misused watermark can be avoided.

For image block A, when its feature watermark matches the extracted watermark, it is considered to have not been tampered. When the feature watermark and the extracted watermark do not match, image block A or B are considered to have been tampered. If image block A is tampered with, it will cause abnormal behavior of the extracted watermark of image block C, with the result that the features of image blocks A and C cannot be matched with the extracted watermark information. On the other hand, if image block A has not been tampered with, the features of image block C and the watermark information can be matched. Thus, by judging the matching of the feature watermark and the extracted watermark of post-offset block C, it is possible to distinguish the tampering condition of image block A. In addition, when it is determined that A is tampered with and image block B is authenticated, the extracted watermark is considered to not have been tampered with, and a restoration can be performed. Otherwise, it is considered that the extracted watermark is not authentic and cannot be restored. The paper is based on this idea of tamper detection and recovery.

3 The algorithm description

3.1 Watermark generation and embedding

Assume the original image as X with size of $M \times N$. Divide it into non-overlapping blocks with size of 2×2 , which can be expressed as $X = (B_1, B_2, \cdots, B_r)$, where $B_i = \begin{pmatrix} b_{i0} & b_{i1} \\ b_{i2} & b_{i3} \end{pmatrix}$ is a 2×2 size image block, and $r = M \times N/4$. The process of watermark generation and embedding can be described as:

Step1. For image block B_i , generate content information B_{Ci} based on key K_1 . The specific method is: Random binary sequences $Z = (z_1, z_2, \cdots, z_{2r})$ are generated based on K_1 . According to the sequences, the 2LSB position of pixel $b_{i(2z_{2i}+z_{2i-1})}$ is set to zero, and the LSBs of all pixels are set to zero.

Step2. Based on key K_2 , the feature watermark W_i of the image block B_i is generated. The specific method is to encrypt the 5 MSBs of the block content B_{Ci} using key K_2 , and generate the feature watermark $W_i = (w_{i1}, w_{i2}, w_{i3}, w_{i4}, w_{i5})$.

Step3. Define $B_{\sigma(i)}$ as the offset block of B_i , where mapping function $\sigma(i)$ needs to meet the following conditions: 1) $\sigma(i) \in \{1,2,\cdots,r\}$, 2) $\sigma(i) \neq i$, and 3) For any $i, j \in \{1,2,\cdots,r\}$, when $i \neq j$ there is $\sigma(i) \neq \sigma(j)$ 。

Step4. Embed the watermark information W_i into the zero setting bits of block $B_{\sigma(i)}$ to generate the watermarked image block $B^w_{\sigma(i)}$.

3.2 Temper detection and recovery

Assume the tampered watermarked image to be $\hat{X}$. Divide it into non-overlapping blocks of size 2×2, represented as $\hat{X}=(\hat{B}_1,\hat{B}_2,\cdots,\hat{B}_r)$. The specific steps of tamper detection and recovery can be described as:

Step1. Based on keys K_1 and K_2, the feature watermark $\hat{W}_i$ of block $\hat{B}_i$ is generated according to the Step1 and Step 2 mentioned above in Section 3.1.

Step2. Get the offset block $\hat{B}_{\sigma(i)}$ of $\hat{B}_i$ using key K_3. Generate the binary sequence $Z=(z_1,z_2,\cdots,z_{2r})$ using K_1, and then obtain the zero setting bits of $\hat{B}_{\sigma(i)}$ from where the embedded watermark W_i' of $\hat{B}_i$ is extracted.

Step3. After obtaining the feature watermark and extracted watermark of all blocks, for a block $\hat{B}_i$, set $\hat{B}_k$ as the pre-offset block. That is, mean $i=\sigma(k)$. According to the matching condition the of feature watermark and extracted watermark, it determines whether the image block $\hat{B}_i$ is tampered with. It can be divided into four cases as follows:

Case1: If $\hat{W}_i=W_i'$, the image block $\hat{B}_i$ is determined as un-tampered.

Case2: If $\hat{W}_i\neq W_i'$ and $\hat{W}_k\neq W_k'$, the image block $\hat{B}_i$ is determined as tampered.

Case3: If $\hat{W}_i\neq W_i'$, $\hat{W}_k=W_k'$ and $\hat{W}_j\neq W_j'$, the image block $\hat{B}_i$ is determined as un-tampered;

Case4: If $\hat{W}_i\neq W_i'$, $\hat{W}_k=W_k'$ and $\hat{W}_j=W_j'$, the image block $\hat{B}_i$ is determined as tampered;

Step4. When block $\hat{B}_i$ is determined as tampered and belongs to Case 2, if $\hat{W}_j=W_j'$, all pixel values in $\hat{B}_i$ can be recovered by decrypting W_i' based on key K_2. Otherwise if $\hat{W}_j\neq W_j'$, the block $\hat{B}_i$ cannot be recovered. When the block $\hat{B}_i$ is determined as tampered and belongs to Case4, replace all pixel values by decrypted W_i' to recover it directly.

4 Performance analysis

4.1 Correlation definitions

For a watermarked image $\hat{X}$, set the rate of tampering is p, that is the proportion of the tampered pixels to all the pixels of the image. For the convenience of description, the following definitions are given firstly.

Definition 1. Assume H_0 as a set of tampered image blocks, and H_1 as a set of un-tempered image blocks. For any block $\hat{B}_i$, define P_{H_0} and P_{H_1} as probabilities of $\hat{B}_i\in H_0$ and $\hat{B}_i\in H_1$, respectively.

According to this definition, there are $H_0\cup H_1=\hat{X}$, $H_0\cap H_1=\phi$. In the case of area tampering, it is considered that all pixels in a block are modified, so

$$\begin{aligned} P_{H_0} &= P\left(\hat{B}_i \in H_0\right) = \frac{|H_0|}{|\hat{X}|} \approx p \\ P_{H_1} &= P\left(\hat{B}_i \in H_1\right) = \frac{|H_1|}{|\hat{X}|} \approx 1 - p \end{aligned} \tag{1}$$

However, in the case of random tampering, the tampered probability of each pixel is p. An image block is considered as tampered if one pixel in it is tampered with, so in this case:

$$\begin{aligned} P_{H_0} &= P\left(\hat{B}_i \in H_0\right) = 1 - (1 - p)^4 \\ P_{H_1} &= P\left(\hat{B}_i \in H_1\right) = (1 - p)^4 \end{aligned} \tag{2}$$

Definition 2. Define $P_{C|H_0}$ as the probability that feature watermark information $\hat{W}_i$ is changed and $P_{L|H_0}$ as the probability that watermark information contained in the LSBs of $\hat{B}_i$ (i.e. the extracting watermark W_k' of block $\hat{B}_k$) in the case of $\hat{B}_i \in H_0$. It is clear that $P_{C|H_0} = 1$. For $P_{C|H_0}$, we need to analyze two cases of area tampering and random tampering:

1) In the case of area tampering, the changed probability of each watermark bit in the LSBs is 1/2,

$$P_{L|H_0} = 1 - \frac{1}{2^5} = \frac{31}{32} \tag{3}$$

2) In the case of random tampering, the probability that x pixels in block $\hat{B}_i$ are tampered with is

$$P_x(\hat{B}_i) = \binom{4}{x} p^x (1 - p)^{4-x} \tag{4}$$

Because the watermark information is embedded in the LSBs and one of the 2LSBs, the probability of the lowest position of $\hat{B}_i$ changed is:

$$P_{x-L}(\hat{B}_i) = \frac{x}{4}(1 - \frac{1}{2^{x+1}}) + \frac{4-x}{4}(1 - \frac{1}{2^x}) \tag{5}$$

So, in case of random tampering:

$$\begin{aligned} P_{L|H_0} &= \sum_{x=1}^{4} P_x(\hat{B}_i) P_{x-L}(\hat{B}_i) \\ &= \sum_{x=1}^{4} \binom{4}{x} p^x (1 - p)^{4-x} \left[\frac{x}{4}(1 - \frac{1}{2^{x+1}}) + \frac{4-x}{4}(1 - \frac{1}{2^x}) \right] \end{aligned} \tag{6}$$

Definition 3. Define $P\{\hat{W}_i \neq W_i'\}$ as the probability that the feature watermark information $\hat{W}_i$ is different with the extracted watermark W_i' for a block $\hat{B}_i$.

So, for any $P\{\hat{W}_i \neq W_i'\}$, according to whether $\hat{W}_i$ and W_i' are changed, it can be divided into four cases and represented by $P_{00}\{\hat{W}_i \neq W_i'\}$, $P_{01}\{\hat{W}_i \neq W_i'\}$, $P_{10}\{\hat{W}_i \neq W_i'\}$, and $P_{11}\{\hat{W}_i \neq W_i'\}$:

1) If both $\hat{W}_i$ and W_i' are not changed:

$$P_{00}\{\hat{W}_i \neq W_i'\} = 0 \tag{7}$$

2) If $\hat{W}_i$ is not changed while W_i' is changed:

$$P_{01}\{\hat{W}_i \neq W_i'\} = 1 \tag{8}$$

3) If $\hat{W}_i$ is changed while W_i' is not changed:

$$P_{10}\{\hat{W}_i \neq W_i'\} = 1 \tag{9}$$

4) If both $\hat{W}_i$ and W_i' are changed, because the length of $\hat{W}_i$ and W_i' are both 5 bits, the probability of $\hat{W}_i = W_i'$ is $1/2^5$, and

$$P_{11}\{\hat{W}_i \neq W_i'\} = 1 - \frac{1}{2^5} = \frac{31}{32} \tag{10}$$

Definition 4. In case of $\hat{B}_i \in H_0$, define $P_{K|H_0}$, $P_{I|H_0}$, $P_{J|H_0}$ as the probabilities of $\hat{W}_k \neq W_k'$, $\hat{W}_i \neq W_i'$, and $\hat{W}_j \neq W_j'$, respectively. In case of $\hat{B}_i \in H_1$, define $P_{K|H_1}$, $P_{I|H_1}$, and $P_{J|H_1}$ as the probabilities of $\hat{W}_k \neq W_k'$, $\hat{W}_i \neq W_i'$, and $\hat{W}_j \neq W_j'$, respectively. According to the four cases in Definition 3, there are

$$\begin{aligned}
P_{K|H_0} &= (1 - P_{C|H_0}P_{H_0})(1 - P_{L|H_0})P_{00}\{\hat{W}_k \neq W_k'\} \\
&+ (1 - P_{C|H_0}P_{H_0})P_{L|H_0}P_{01}\{\hat{W}_k \neq W_k'\} \\
&+ P_{C|H_0}P_{H_0}(1 - P_{L|H_0})P_{10}\{\hat{W}_k \neq W_k'\} \\
&+ P_{C|H_0}P_{H_0}P_{L|H_0}P_{11}\{\hat{W}_k \neq W_k'\} \\
&= (1 - P_{C|H_0}P_{H_0})P_{L|H_0} + P_{C|H_0}P_{H_0}(1 - P_{L|H_0}) \\
&+ P_{C|H_0}P_{H_0}P_{L|H_0}\frac{31}{32} \\
&= P_{L|H_0} + P_{C|H_0}P_{H_0} - \frac{33}{32}P_{C|H_0}P_{L|H_0}P_{H_0}
\end{aligned} \tag{11}$$

$$
\begin{aligned}
P_{K|H_1} &= (1 - P_{C|H_0} P_{H_0})(1 - P_{L|H_1}) P_{00}\{\hat{W}_k \neq W'_k\} \\
&+ (1 - P_{C|H_0} P_{H_0}) P_{L|H_1} P_{01}\{\hat{W}_k \neq W'_k\} \\
&+ P_{C|H_0} P_{H_0}(1 - P_{L|H_1}) P_{10}\{\hat{W}_k \neq W'_k\} \\
&+ P_{C|H_0} P_{H_0} P_{L|H_1} P_{11}\{\hat{W}_k \neq W'_k\} \\
&= (1 - P_{C|H_0} P_{H_0}) P_{L|H_1} + P_{C|H_0} P_{H_0}(1 - P_{L|H_1}) + P_{C|H_0} P_{H_0} P_{L|H_1} \frac{31}{32} \\
&= P_{C|H_0} P_{H_0}
\end{aligned} \tag{12}
$$

$$
\begin{aligned}
P_{I|H_0} &= (1 - P_{C|H_0})(1 - P_{L|H_0} P_{H_0}) P_{00}\{\hat{W}_i \neq W'_i\} \\
&+ (1 - P_{C|H_0}) P_{L|H_0} P_{H_0} P_{01}\{\hat{W}_i \neq W'_i\} \\
&+ P_{C|H_0}(1 - P_{L|H_0} P_{H_0}) P_{10}\{\hat{W}_i \neq W'_i\} \\
&+ P_{C|H_0} P_{L|H_0} P_{H_0} P_{11}\{\hat{W}_i \neq W'_i\} \\
&= (1 - P_{C|H_0}) P_{L|H_0} P_{H_0} + P_{C|H_0}(1 - P_{L|H_0} P_{H_0}) \\
&+ P_{C|H_0} P_{L|H_0} P_{H_0} \frac{31}{32} \\
&= P_{L|H_0} P_{H_0} + P_{C|H_0} - \frac{33}{32} P_{C|H_0} P_{L|H_0} P_{H_0}
\end{aligned} \tag{13}
$$

$$
\begin{aligned}
P_{I|H_1} &= (1 - P_{C|H_1})(1 - P_{L|H_0} P_{H_0}) P_{00}\{\hat{W}_i \neq W'_i\} \\
&+ (1 - P_{C|H_1}) P_{L|H_0} P_{H_0} P_{01}\{\hat{W}_i \neq W'_i\} \\
&+ P_{C|H_1}(1 - P_{L|H_0} P_{H_0}) P_{10}\{\hat{W}_i \neq W'_i\} \\
&+ P_{C|H_1} P_{L|H_0} P_{H_0} P_{11}\{\hat{W}_i \neq W'_i\} \\
&= (1 - P_{C|H_1}) P_{L|H_0} P_{H_0} + P_{C|H_1}(1 - P_{L|H_0} P_{H_0}) \\
&+ P_{C|H_1} P_{L|H_0} P_{H_0} \frac{31}{32} \\
&= P_{L|H_0} P_{H_0}
\end{aligned} \tag{14}
$$

$$
\begin{aligned}
P_{J|H_0} &= P_{J|H_1} = (1 - P_{C|H_0}P_{H_0})(1 - P_{L|H_0}P_{H_0})P_{00}\{\hat{W}_j \neq W'_j\} \\
&+ (1 - P_{C|H_0}P_{H_0})P_{L|H_0}P_{H_0}P_{01}\{\hat{W}_j \neq W'_j\} \\
&+ P_{C|H_0}P_{H_0}(1 - P_{L|H_0}P_{H_0})P_{10}\{\hat{W}_j \neq W'_j\} \\
&+ P_{C|H_0}P_{H_0}P_{L|H_0}P_{H_0}P_{11}\{\hat{W}_j \neq W'_j\} \\
&= (1 - P_{C|H_0}P_{H_0})P_{L|H_0}P_{H_0} + P_{C|H_0}P_{H_0}(1 - P_{L|H_0}P_{H_0}) \\
&+ P_{C|H_0}P_{H_0}P_{L|H_0}P_{H_0}\frac{31}{32} \\
&= P_{L|H_0}P_{H_0} + P_{C|H_0}P_{H_0} - \frac{33}{32}P_{C|H_0}P_{L|H_0}P_{H_0}^2
\end{aligned} \tag{15}
$$

4.2 Ability of tamper detection

In this section, we will analyze the ability to detect tampering by analyzing the false positive rate of the four cases in Section 3.2.

For **Case 1**, it is false positive if $\hat{W}_i = W'_i$ when $\hat{B}_i$ is a tampered block, and its false positive rate is:

$$
\begin{aligned}
P_{f1} &= P\{\hat{W}_i = W'_i \mid \hat{B}_i \in H_0\} \\
&= 1 - P\{\hat{W}_i \neq W'_i \mid \hat{B}_i \in H_0\} \\
&= 1 - P_{I|H_0}
\end{aligned} \tag{16}
$$

For **Case 2**, it is false positive if $\hat{W}_i \neq W'_i$ and $\hat{W}_k \neq W'_k$ when $\hat{B}_i$ is an un-tampered block, and its false positive rate is:

$$
\begin{aligned}
P_{f2} &= P\{\hat{W}_i \neq W'_i \cap \hat{W}_k \neq W'_k \mid \hat{B}_i \in H_1\} \\
&= P\{\hat{W}_i \neq W'_i \mid \hat{B}_i \in H_1\}P\{\hat{W}_k \neq W'_k \mid \hat{B}_i \in H_1\} \\
&= P_{I|H_1}P_{K|H_1}
\end{aligned} \tag{17}
$$

For **Case 3**, it is false positive if $\hat{W}_i \neq W'_i$, $\hat{W}_k = W'_k$ and $\hat{W}_j \neq W'_j$ when $\hat{B}_i$ is an tampered block, and its false positive rate is:

$$
\begin{aligned}
P_{f3} &= P\{\hat{W}_i \neq W'_i \cap \hat{W}_j \neq W'_j \cap \hat{W}_k = W'_k \mid \hat{B}_i \in H_0\} \\
&= P\{\hat{W}_i \neq W'_i \mid \hat{B}_i \in H_0\}P\{\hat{W}_j \neq W'_j \mid \hat{B}_i \in H_0\}P\{\hat{W}_k = W'_k \mid \hat{B}_i \in H_0\} \\
&= P_{I|H_0}P_{J|H_0}(1 - P_{K|H_0})
\end{aligned} \tag{18}
$$

For **Case 4**, it is false positive if $\hat{W}_i \neq W_i'$, $\hat{W}_j = W_j'$ and $\hat{W}_k = W_k'$ when $\hat{B}_i$ is an un-tampered block, and its false positive rate is:

$$\begin{aligned}
P_{f4} &= P\{\hat{W}_i \neq W_i' \cap \hat{W}_j = W_j' \cap \hat{W}_k = W_k' \mid \hat{B}_i \in H_1\} \\
&= P\{\hat{W}_i \neq W_i' \mid \hat{B}_i \in H_1\} P\{\hat{W}_j = W_j' \mid \hat{B}_i \in H_1\} P\{\hat{W}_k = W_k' \mid \hat{B}_i \in H_1\} \\
&= P_{I|H_1}(1 - P_{J|H_1})(1 - P_{K|H_1})
\end{aligned} \tag{19}$$

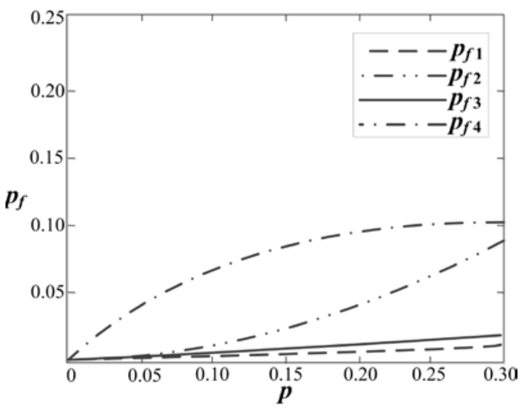


Figure 3: False positive rate with tamper rate in various cases under area tampering

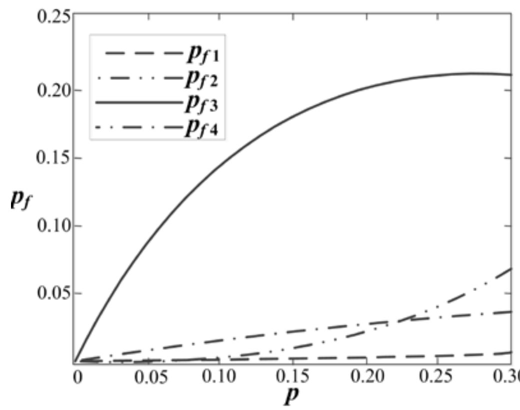


Figure 4: False positive rate with tamper rate in various cases under random tampering

Fig. 3 shows the values of P_{f1}, P_{f2}, P_{f3} and P_{f4} with the rate of tampering p from 0-30% under the area tampering. The false positive rates of all kinds of cases are below 10%, indicating that the tamper situation can be accurately determined under the area tampering. Fig. 4 gives the values of P_{f1}, P_{f2}, P_{f3} and P_{f4} with the rate of tampering p from 0-15%, under random tampering. It can be seen from the figure that the false positive rate of Case 3 is maximum, while the other three cases put up smaller false positive rate. This indicates that these three cases can accurately determine whether the tampered image blocks under random tampering.

4.3 Missing detection rate and false detection rate

According to Section 3.2, it is missing detection when Case 1 and Case 3 are false positive:

$$P_{fa} = P_{f1} + P_{f3} \tag{20}$$

It is false detection when Case 1 and Case 3 are false positive:

$$P_{fr} = P_{f2} + P_{f4} \tag{21}$$

The missing detection rate (the missing detected blocks account for the proportion of all image blocks) and false detection rate (the false detected blocks account for the proportion of all image blocks) are, respectively:

$$R_{fa} = P_{fa} \times P_{H_0} \quad R_{fr} = P_{fr} \times P_{H_1} \tag{22}$$

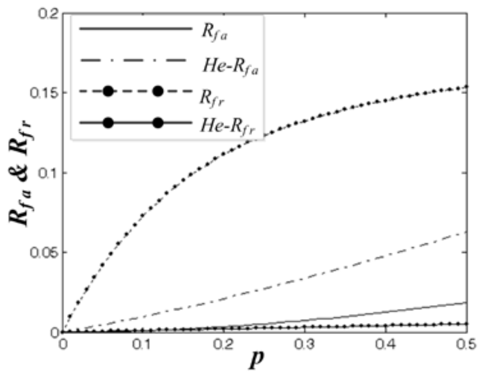


Figure 5: Missing detection rate and false detection rate with tamper rate under area tampering

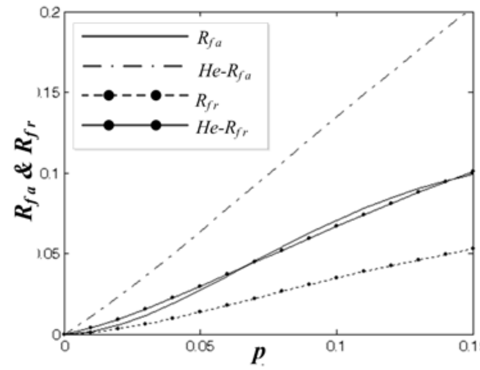


Figure 6: Missing detection rate and false detection rate with tamper rate under random tampering

Fig. 5 shows the missing detection rate and false detection rate comparison of proposed algorithm and the algorithm of He et al. [He, Chen, Tai et al. (2012)] (named He-Rfa and He-Rfr respectively) under area tampering. Under area tampering, the false detection rate of the proposed algorithm is higher than that of He et al. [He, Chen, Tai et al. (2012)], but the missing detection rate is obviously lower than that of [He, Chen, Tai et al. (2012)]. Fig. 6 shows the missing detection rate and false detection rate comparison of the proposed algorithm and the algorithm of He et al. [He, Chen, Tai et al. (2012)] under random tampering. Under random tampering, the false detection rate and missing detection rate of the proposed algorithm are both lower than that of He et al. [He, Chen, Tai et al. (2012)]. This indicates that the proposed algorithm can detect the random tampered image blocks more accurately.

5 Experiments and analysis

We used the 256×256 grayscale images shown in Fig. 7 as the test, with pixels between [0~255]. The chaotic scrambling method is used to select the offset blocks. In the authentication results, the non-tampered image blocks are represented by black (gray value: 0), and the tampered image blocks are represented by white (gray value: 255). The validity of the proposed algorithm is verified from two aspects: Tamper detection and recovery capability, resistance to random tamper ability test.

Barbara Lena Benz Pepper

Figure 7: Test images

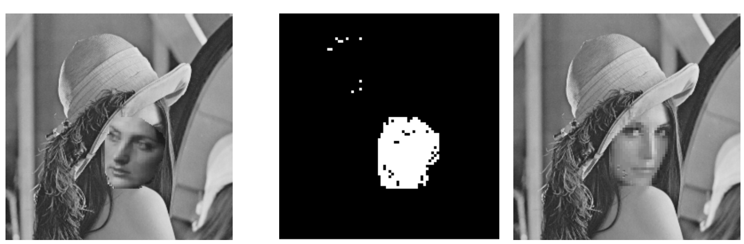
5.1 Tamper detection and recovery capability

To verify the tamper detection and recovery abilities of the proposed algorithm, we verify its effectiveness in a clipping attack, image collage attack, intra-image collage attack, and so on.

Single region clipping | Detection result | Recovery result

Multiple regions clipping | Detection result | Recovery result

Figure 8: Temper detection and recovery results for clipping attack



Collage attack of two images | Detection result | Recovery result

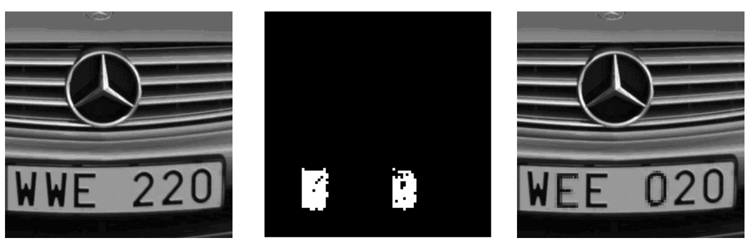
Collage attack in the same region of the two images | Detection result | Recovery result

Figure 9: Tamper detection and recovery results for collage attack between images

Fig. 8 shows the temper detection and recovery result images for a clipping attack. The single region and multiple regions of the clipping image are tested. From the experimental results, we can see that the algorithm can detect the clipping area accurately and carry out a high-quality recovery.

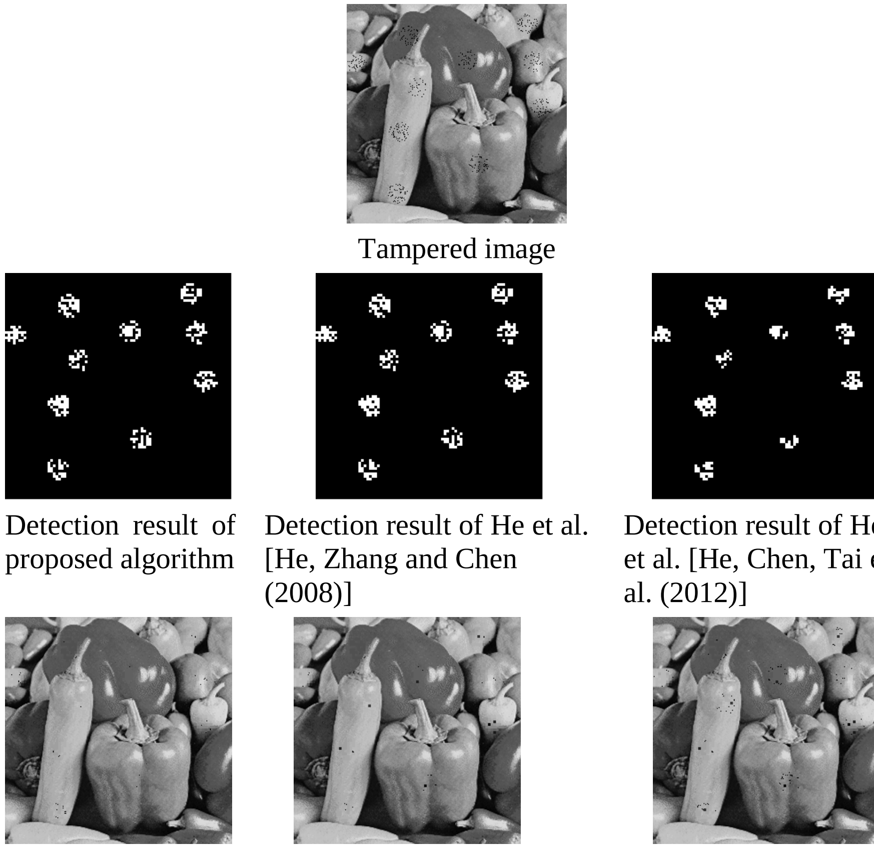
Fig. 9 shows the tamper detection and recovery result images for a collage attack between images. The collage attack of two images and the collage attack in the same region of the two images are tested. From the experimental results, we can see that the algorithm can not only detect the collage area accurately, but also detect and recover the collage attack in the same region.

Fig. 10 shows the temper detection and recovery result images for a collage attack inside an image. In it, the first letter “W” of the license plate is copied onto the second letter “E”, and the second number “2” is copied to the first number “0”, so that the license plate number is modified to “WWE 220”. It can be seen from the experimental results that the proposed algorithm can accurately detect the tampered regions of intra-image collage and carry out high-quality recovery.



Collage attack internal an image | Detection result | Recovery result

Figure 10: Temper detection and recovery results for a collage attack internal an image



Tampered image

Detection result of proposed algorithm

Detection result of He et al. [He, Zhang and Chen (2008)]

Detection result of He et al. [He, Chen, Tai et al. (2012)]

Recovery result of proposed algorithm

Recovery result of He et al. [He, Zhang and Chen (2008)]

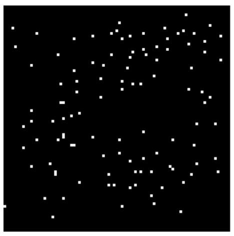
Recovery result of He et al. [He, Chen, Tai et al. (2012)]

Figure 11: Detection and recovery results under random tampering with "spray gun"

5.2 Detection ability to random tampering

For random tamper attacks, the "Pepper" image is random tampered with a "spray gun" image and "pencil" tools in Windows Drawing. To verify the effectiveness of the proposed algorithm, it is compared with the neighborhood-based authentication algorithm (see He et al. [He, Zhang and Chen (2008); He, Chen, Tai et al. (2012)]). Fig. 11 shows the result of detection and recovery for the image tampered by the "spray gun" tool. Fig. 12 uses the "pencil" tool to make random single pixel tampering of the watermarked image, and its authentication and recovery results. From the experimental results, we can see that, compared with the algorithms of He et al. [He, Zhang and Chen (2008); He, Chen, Tai et al. (2012)], the proposed algorithm can detect the areas of random tampering and even single pixel tampering accurately, and achieve high-quality recovery.

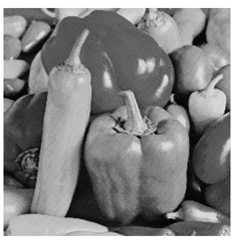
Tampered image



Detection result of proposed algorithm

Detection result of He et al. [He, Zhang and Chen (2008)]

Detection result of He et al. [He, Chen, Tai et al. (2012)]



Recovery result of proposed algorithm

Recovery result of He et al. [He, Zhang and Chen (2008)]

Recovery result of He et al. [He, Chen, Tai et al. (2012)]

Figure 12: Detection and recovery results under random tampering with “pencil”

6 Conclusions

Self-embedded fragile watermarking technology can detect and recover the tampered regions of an image and has received extensive attention and research. In this paper, a self-embedding fragile watermarking algorithm based on combined decision using offset blocks is proposed to solve the problem of accurate detection and recovery under random tampering. The main contributions of this article include the following:

1. In this paper, the feature of the image block is generated as the watermark, and is used not only for tamper detection, but also for recovery. Thus, the length of watermark is reduced.

2. Image block tampering detection is done by matching the feature watermark and extracted watermark from the pre-offset and post-offset blocks. This makes it avoid the false detection caused by the extracted watermark being tampered with and does not need the neighborhood image block which can effectively detect randomly tampered image.

3. When the tampering is reversed, the recovery is improved by judging whether the post-offset block is tampered with or not, and the credibility of the recovery is improved.

However, the proposed algorithm is only applicable to the authentication and recovery of space image. How to adapt the fragile watermarking algorithm for compressed image is the main research direction of this paper.

Acknowledgment: This work was supported in part by the National Natural Science Foundation of China (No. 61401512，61602508, 61772549, 6141512 and U1636219), the National Key R&D Program of China (No. 2016YFB0801303 and 2016QY01W0105), the Key Technologies R&D Program of Henan Province (No. 162102210032), and the Key Science and Technology Research Project of Henan Province (No. 152102210005).

References

Cao, F.; An, B.; Wang, J.; Ye, D.; Wang, H. (2017): Hierarchical recovery for tampered images based on watermark self-embedding. *Displays*, vol. 2017, no. 46, pp. 52-61.

Chen, F.; He, H. J.; Huo, Y.; Wang, H. (2011): Self-recovery fragile watermarking scheme with variable watermark payload*. Digital Forensics and Watermarki*ng, pp. 142-155.

Chen, F.; He, H. J.; Wang, H. X. (2012): Variable-payload self-recovery watermarking scheme for digital image authentication. *Chinese Journal of Computers*, vol. 35, no. 1, pp. 154-162.

Chang, C. C.; Fan, Y. H.; Tai, W. L. (2008): Four-scanning attack on hierarchical digital watermarking method for image tamper detection and recovery. *Pattern Recognition*, vol. 41, no. 2, pp. 654-661.

Chang, Y. F.; Tai, W. L. (2013): A block-based watermarking scheme for image tamper detection and self-recovery. *Opto-Electronics Review*, vol. 21, no. 2, pp. 182-190.

Deng, X.; Chen, Z.; Zeng, F.; Zhang, Y.; Mao, Y. (2013): Authentication and recovery of medical diagnostic image using dual reversible digital watermarking. *Journal of Nanoscience and Nanotechnology*, vol. 13, no. 3, pp. 2099-2107.

Dhole, V. S.; Patil, N. N. (2015): Self embedding fragile watermarking for image tampering detection and image recovery using self-recovery blocks. *International Conference on Computing Communication Control and Automation*, pp. 752-757.

Duan, G. D.; Zhao, X.; Li, J. P.; Liao, J. M. (2010): A novel semi-fragile digital watermarking algorithm for image content authentication, localization and recovery. *Acta Electronica Sinica*, vol. 38, no. 4, pp. 842-847.

He, H. J.; Chen, F. (2007): On the security of the self-embedding watermarking scheme. *Acta Electronica Sinica*, vol. 35, no. 3, pp. 557-562.

He, H. J.; Chen, F.; Tai, H. M.; Kalker, T.; Zhang, J. (2012): Performance analysis of a block-neighborhood-based self-recovery fragile watermarking scheme. *IEEE Transactions on Information Forensics & Security*, vol. 7, no. 1, pp. 185-196.

He, H. J.; Zhang, J. S.; Chen, F. (2008): A self-recovery fragile watermarking scheme for image authentication with superior localization. *Science in China Series F:*

Information Sciences, vol. 51, no. 10, pp. 1487-1507.

Kiatpapan, S.; Kondo, T. (2015): An image tamper detection and recovery method based on self-embedding dual watermarking. *International Conference on Electrical Engineering/electronics, Computer, Telecommunications and Information Technology*, pp. 1-6.

Lin, P. L.; Hsieh, C. K.; Huang, P. W. (2005): A hierarchical digital watermarking method for image tamper detection and recovery. *Pattern Recognition*, vol. 38, no. 12, pp. 2519-2529.

Qin, C.; Ji, P.; Wang, J.; Chang, C. C. (2017): Fragile image watermarking scheme based on VQ index sharing and self-embedding. *Multimedia Tools & Applications*, vol. 76, no. 2, pp. 2267-2287.

Singh, D.; Singh, S. K. (2016): *Effective Self-embedding Watermarking Scheme for Image Tampered Detection and Localization with Recovery Capability*. Academic Press, USA.

Wang, G. D.; Liu, F. L.; Liu, Y.; Yao, G. (2008): An image authentication scheme with discrimination of tampers on watermark or image. *Acta Electronica Sinica*, vol. 36, no. 7, pp. 1349-1354.

Yang, W. M.; Cai, J. (2011): Image fragile watermarking algorithm of self-embedding. *Journal of Chinese Computer Systems*, vol. 32, no. 1, pp. 169-172.