

# A new semi-fragile image watermarking with robust tampering restoration using irregular sampling

Xunzhan Zhu<sup>a,\*</sup>, Anthony T.S. Ho<sup>b</sup>, Pina Marziliano<sup>a</sup>

<sup>a</sup>*School of Electrical and Electronic Engineering, Nanyang Technological University, 639798, Singapore*

<sup>b</sup>*School of Electronics and Physical Sciences, University of Surrey, GU2 7XH, UK*

Received 19 April 2005; received in revised form 26 March 2007; accepted 28 March 2007

---

## Abstract

This paper presents a semi-fragile watermarking method for the automatic authentication and restoration of the content of digital images. Semi-fragile watermarks are embedded into the original image, which reflect local malicious tampering on the image. When tampered blocks are detected, the restoration problem is formulated as an irregular sampling problem. These blocks are then reconstructed, making use of the information embedded in the same watermarked image, through iterative projections onto convex sets. In contrast to previous methods, the restoration process is robust to common image processing operations such as lossy transcoding and image filtering. Simulation results showed that the scheme keeps the probability of false alarm to a minimum while maintaining the data integrity of the restored images.

© 2007 Elsevier B.V. All rights reserved.

*Keywords:* Authentication; Digital watermarking; Irregular sampling; Projection onto convex sets; Restoration

---

## 1. Introduction

Digital photographs and videos are widely used nowadays; however, warnings about the potential for faking digital images are still present. There is a growing need for techniques which could provide some form of assurance that the image has not been tampered with. In certain practical applications, such as remote sensing, legal defending, news reporting and medical archiving, it is desirable for some initial estimated reconstruction of the tampered parts of the image content.

Many fragile or semi-fragile watermarking methods have been proposed for image content authentication [1–4,10,11,16,18], in which a hidden message or watermark is embedded into the original image and later used to detect changes to the watermarked image. The fragile watermarking schemes are designed to detect any slight changes to the bits of the watermarked image and the watermark becomes undetectable after the watermarked image is modified in any way. The semi-fragile watermarking seeks to verify that the content of the multimedia has not been modified by illegitimate distortions, while allowing modification by legitimate distortions [5]. The approaches for automatic reconstruction of tampered areas have also been found in some schemes. In [6], Fridrich and Goljan proposed two self-embedding schemes

---

\*Corresponding author.

*E-mail addresses:* [xzzhu@pmail.ntu.edu.sg](mailto:xzzhu@pmail.ntu.edu.sg) (X. Zhu), [a.ho@surrey.ac.uk](mailto:a.ho@surrey.ac.uk) (A.T.S. Ho), [epina@ntu.edu.sg](mailto:epina@ntu.edu.sg) (P. Marziliano).

for the self-restoration of digital images with watermarking authentication.

The first scheme began with dividing the image into  $8 \times 8$  blocks and transforming each block using DCT. The primary DCT coefficients of every block were then quantized, using a quantization matrix corresponding to that of 50% quality JPEG compression. The resulting bit-string for every block was carefully controlled so that it was exactly 64-bit long. These bits were then embedded into the least significant bits (LSBs) of another block. When the watermarked content was detected to be unauthentic, the recovery bits were extracted out to establish a low-resolution reconstruction of the modified parts. However, the quality of the recovery was not satisfactory and may not be sufficient for illustrating fine details. To improve the quality of the reconstruction, the authors proposed extending the recovery bits to be 128 bits for every block, and using two LSBs for embedding. However, this introduced more degradation to the watermarked images. Moreover, the spatial domain based method was easily attacked. The restoration would fail if the attacker, after modifying the content, further compresses the image file or just randomizes the least significant bits.

The second scheme encoded the information of the original image into the watermarked image using differential encoding. First, a low color depth image  $\mathbf{t}$  was generated by decreasing the gray levels of the original image  $\mathbf{g}$  to the interval  $[-8, 8]$ . A cyclic shift on  $\mathbf{t}$  was then performed by an integer vector  $(a, b)$ :

$$\tilde{\mathbf{t}}_{ij} = \mathbf{t}_{i_a j_b},$$

where

$$i_a = (i - a) \bmod M,$$

$$j_b = (j - b) \bmod N.$$

The watermark embedding process started from the upper left corner and proceeded by rows from left to right, top to down. The watermarked image  $\mathbf{g}'$  was defined as

$$\mathbf{g}'_{11} = \mathbf{g}_{11},$$

$$\bmod(\mathbf{g}'_{ij+1} - \mathbf{g}'_{ij}) = \tilde{\mathbf{t}}_{ij}.$$

In the reconstruction stage, a color truncated approximation to the original image can be obtained by calculating the difference between pixels. This improved scheme can tolerate the least

amount of image processing operations, such as JPEG compression with quality factor above 85% and additive random noise in the range  $[-2, 2]$ .

Alternative models for blind restoration incorporating watermarking have been proposed such as the telltale watermarking by Kundur et al. [10]. Such work is designed to invert the distortions on the watermarked images, based on the assumption that the distortions can be approximately determined and is invertible. However, the telltale watermarking is not useful for restoration of removal attacks since they are not invertible.

In this paper, we propose a novel semi-fragile watermarking method which is capable of robust self-restoration by casting it as an irregular sampling problem. The restoration is then performed by projections onto convex sets [12]. The watermark signal is generated by combining a pseudo-random signal with some prior knowledge of the carrier image, which belongs to a convex set. This mixed signal is then embedded into the original image. When tampered areas are detected, they are restored by iterative projections onto the convex sets. Experimental results showed that accurate tamper detection and restoration were still possible after lossy transcoding and other common image processing operations.

The rest of this paper is organized as follows. In Section 2 we introduce our proposed approach. The simulation results and performance analysis are reported in Section 3, followed by concluding remarks and future work in Section 4.

## 2. Proposed watermarking method

This section describes the proposed watermarking method. A brief review of image restoration from irregular samples by projections onto convex sets is given in Section 2.1. The watermark embedding process is discussed in Section 2.2. In Section 2.3, we describe the processes of image authentication and restoration. The minimization of the probability of false alarm is addressed in Section 2.4. Finally, the security issue is addressed in Section 2.5.

### 2.1. Restoration from irregular samples by projections onto convex sets

In this paper, we address the problem of detecting tampered blocks in a watermarked image and then restoring them. Assume that we have no prior knowledge of the corruption channel, the problem

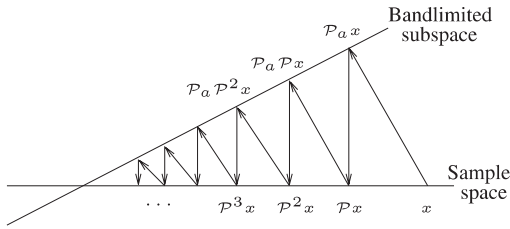


Fig. 1. Projection onto convex sets.  $\mathcal{P} = \mathcal{P}_b \mathcal{P}_a$ , where  $\mathcal{P}_a$  is the first projection onto a bandlimited subspace and  $\mathcal{P}_b$  is the second projection onto the space of unknown samples.

can be formulated as obtaining an incomplete set of data with lost packets, which can be cast as an irregular sampling problem that can be solved by projection onto convex<sup>1</sup> sets (POCS) method [12,17] under the assumption that the signal belongs to two linear convex sets with non-empty intersection. It involves two projections: the first projection is onto a band-limited subspace  $l_a$  and the second projection is onto the space of unknown samples  $l_b$ , as exemplified in Fig. 1, where  $\mathcal{P} = \mathcal{P}_b \mathcal{P}_a$ , and  $\mathcal{P}_a$  and  $\mathcal{P}_b$  are the projection operators onto the two convex sets, respectively.

To facilitate the watermarking based restoration problem, we define the following two convex sets:

(1)  $l_a$  denotes the subset of  $\mathcal{H}$ , where  $\mathcal{H}$  is a Hilbert space, composed of all functions whose cosine transform coefficients satisfy the constraint

$$\text{sgn}(F(u, v)) = \Gamma(u, v) \tag{1}$$

in a prescribed region  $\Delta$  of the frequency domain, where  $F(u, v)$  is the DCT coefficient of  $f(x, y)$ ,  $\Gamma(u, v) \in \{0, 1\}$  is a known binary function, and

$$\text{sgn}(y) = \begin{cases} 1, & y \geq 0, \\ 0, & y < 0. \end{cases}$$

The projection<sup>2</sup> of an arbitrary  $f \in \mathcal{H}$  onto  $l_a$  is realized by

$$\mathcal{P}_a f \leftrightarrow \begin{cases} F(u, v), & (u, v) \in \Delta, \text{sgn}(F(u, v)) = \Gamma(u, v), \\ 0, & (u, v) \in \Delta, \text{sgn}(F(u, v)) \neq \Gamma(u, v), \\ F(u, v), & (u, v) \notin \Delta. \end{cases} \tag{2}$$

<sup>1</sup>A subset  $l$  of  $\mathcal{H}$ , where  $\mathcal{H}$  is a Hilbert space, is said to be convex if together with any  $x_1$  and  $x_2$  it also contains  $\mu x_1 + (1 - \mu)x_2$  for all  $\mu, 0 \leq \mu \leq 1$ .

<sup>2</sup>Given  $f \in \mathcal{H}$ , the projection  $g \triangleq \mathcal{P}_i f$  of  $f$  onto the closed convex set  $l_i$  is that unique element  $g \in l_i$  that satisfies

$$\inf_{x \in l_i} \|f - x\| = \|f - g\|,$$

where  $\|x\| \triangleq \langle x, x \rangle^{1/2}$  is the norm in the  $L_2$ -space.

During watermarking, the polarity information of the primary cosine transform coefficients is extracted, hashed into the watermark signal, and embedded into the image itself. This prior knowledge is extracted at the detection phase and used to restore the tampered blocks.

(2)  $l_b$  denotes the set of all functions in  $\mathcal{H}$  which assume prescribed values  $\Theta$  over a closed region  $\Delta$ . The projection onto  $l_b$  is realized by

$$\mathcal{P}_b f = \begin{cases} \Theta(x, y), & (x, y) \in \Delta, \\ f(x, y), & (x, y) \notin \Delta. \end{cases} \tag{3}$$

Here,  $\Theta(x, y)$  are the values of the known samples. The convexity and closure of the above sets can be proved. Let us define

$$\mathcal{P} = \mathcal{P}_b \mathcal{P}_a, \tag{4}$$

then the signal can be restored through the iteration

$$f^{(i+1)} = \mathcal{P} f^{(i)} \tag{5}$$

with  $f^{(0)}$  being the initial irregularly sampled signal.

## 2.2. Watermark embedding

In [7], a semi-fragile watermarking method for image authentication was proposed in the pinned sine transform (PST) domain. In PST, an image field was decomposed into two sub-fields, i.e., the boundary field and a residual field [14], known as the pinned field which vanished at the boundaries. For the pinned field, its Karhunen–Loeve transform (KLT) is the sine transform. We found that the pinned field is a good characterization of edges, which largely reflects the texture information in the original image. The watermark was embedded into the sine transform domain of the pinned field as an indicator of the authenticity of the watermarked image. Since most common image manipulations tend to preserve such primary features of images, this embedding method ensures that the watermark does not suffer significantly from legitimate manipulations, such as compression and some common system processing operations.

The process of watermark embedding begins with dividing the original image into sub-blocks of size  $n \times n$ . These sub-blocks are then grouped into macro-blocks which contain  $m \times m$  sub-blocks in raster order. In general, the authentication is based on sub-blocks while the restoration is based on macro-blocks. In the following, the words “block” and “sub-block” will be used alternatively. Consider

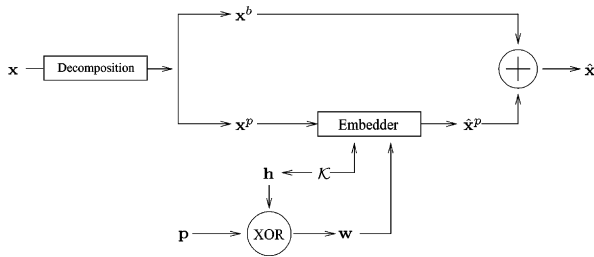


Fig. 2. Watermark embedding process for one sub-block.

re-ordering in zigzag scan the DCT coefficients of the macro-block  $\mathbf{X}_\mu$ . If  $\ell$  is the number of watermark bits to be embedded into every  $n \times n$  sub-block, then define  $\mathbf{d}_\mu$  as the first  $\ell m^2$  coefficients in this re-ordered coefficient set. The polarity information,  $\mathbf{p}_\mu$ , is generated by

$$p_\mu(k) = \begin{cases} 1, & d_\mu(k) \geq 0, \\ 0, & d_\mu(k) < 0, \end{cases} \quad (6)$$

where  $k = 0, 1, \dots, \ell m^2 - 1$ . The macro-blocks are then formed into pairs using a pre-determined mapping function  $\Omega$ . Suppose  $\Omega(\mu) = \nu$ , where  $\nu$  is the index of another macro-block, then  $\mathbf{p}_\mu$ , the polarity information of macro-block  $\mathbf{X}_\mu$  is to be embedded into  $\mathbf{X}_\nu$ , and  $\mathbf{p}_\nu$  is to be embedded into  $\mathbf{X}_\mu$ , with  $\ell$  bits into each  $n \times n$  block. A pseudo-random binary signal  $\mathbf{h}$  is generated and its initial state is contained as part of the secret key file  $\mathcal{K}$ . The watermark signal  $\mathbf{w}_\mu$  is then obtained by XORing the pseudo-random signal with the polarity information:

$$\mathbf{w}_\mu = \mathbf{h} \oplus \mathbf{p}_\mu. \quad (7)$$

The watermark is partitioned into  $m^2$  parts, and each part is embedded into individual sub-blocks.

Fig. 2 describes the watermark embedding process for one sub-block. Consider an  $n \times n$  block  $\mathbf{x}$ , it is first decomposed into two fields,<sup>3</sup> the boundary field  $\mathbf{x}^b$  and the pinned field  $\mathbf{x}^p$ , which can be described as

$$\mathbf{x} = \mathbf{x}^b + \mathbf{x}^p. \quad (8)$$

Next, the sine transform is applied to the pinned field block as follows:

$$\mathbf{x}^{p(s)} = \mathbf{S}_n \mathbf{x}^p \mathbf{S}_n^T, \quad (9)$$

<sup>3</sup>Refer to Eqs. (15)–(24) for the specific process.

where  $\mathbf{S}_n$  is the sine transform matrix of order  $n$  [9]:

$$S_n(i, j) = \sqrt{\frac{2}{n+1}} \sin \frac{\pi(i+1)(j+1)}{n+1}, \quad (10)$$

where  $0 \leq i, j \leq n-1$ .

In the middle to high frequency bands of  $\mathbf{x}^{p(s)}$ , we select  $\ell$  coefficients for watermarking modulation according to the length of the watermark signal. A specific bit  $w(k)$  is embedded into a coefficient  $x^{p(s)}(k)$  according to the following algorithm:

**Algorithm 1.** Watermark embedding.

```

if  $w(k) = 1$  then
  if  $x^{p(s)}(k) > \lambda$  then
     $\hat{x}^{p(s)}(k) = x^{p(s)}(k)$ 
  else
     $\hat{x}^{p(s)}(k) = \alpha_1$ 
  end if
else if  $w(k) = 0$  then
  if  $x^{p(s)}(k) < -\lambda$  then
     $\hat{x}^{p(s)}(k) = x^{p(s)}(k)$ 
  else
     $\hat{x}^{p(s)}(k) = \alpha_2$ 
  end if
end if

```

where:

- $\hat{x}^{p(s)}(k)$  is the corresponding watermarked coefficient.
- $\lambda$  is a sufficiently large threshold of positive value. It can be determined by users; its value will affect the tradeoff between the perceptual quality of the watermarked image and the robustness of the semi-fragile watermark.
- $\alpha_1$  and  $\alpha_2$  are floating point values chosen randomly from  $[\lambda/2, \lambda]$  and  $[-\lambda, -\lambda/2]$ , respectively.

The watermarked pinned field block  $\hat{\mathbf{x}}^p$  is obtained by the inverse 2D sine transform:

$$\hat{\mathbf{x}}^p = \mathbf{S}_n^T \hat{\mathbf{x}}^{p(s)} \mathbf{S}_n \quad (11)$$

and a watermarked block is therefore achieved by

$$\hat{\mathbf{x}} = \mathbf{x}^b + \hat{\mathbf{x}}^p. \quad (12)$$

### 2.3. Image authentication and self-restoration

The watermark detection and image authentication processes for one sub-block are illustrated in Fig. 3. The detection system receives as input a

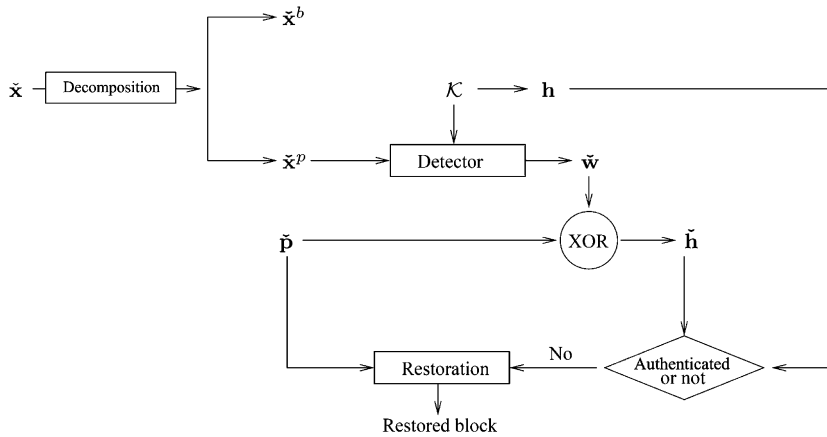


Fig. 3. Watermark detection, image authentication and restoration for one sub-block.

watermarked and possibly tampered image. Similar to the watermarking process, the polarity information is extracted from every macro-block and is partitioned into  $m^2$  parts, with every part corresponding to one sub-block in its paired macro-block. Here, we assume that the pre-determined mapping function  $\Omega$  is known to both encoder and decoder. Since we limit ourselves to the situations in which tampering is only in the form of content modification, the synchronization issue after geometrical attacks is not considered here.

The embedded watermark is extracted from every sub-block by the following algorithm:

**Algorithm 2.** Watermark detection.

```

if  $\tilde{x}^{p(s)}(k) \geq 0$  then
     $\tilde{w}(k) = 1$ 
else
     $\tilde{w}(k) = 0$ 
end if
    
```

where  $\tilde{w}$  is the extracted watermark. The extracted watermark is then XOR-ed with the corresponding part of the polarity information of its paired macro-block:

$$\tilde{\mathbf{h}} = \tilde{\mathbf{w}} \oplus \check{\mathbf{p}}. \tag{13}$$

The original pseudo-random signal  $\mathbf{h}$  is also generated using the initial state in  $\mathcal{K}$ . The bits in  $\tilde{\mathbf{h}}$  and  $\mathbf{h}$  are then compared by the normalized cross correlation function  $\rho$ , whose value lies in  $[0, 1]$ . Assume  $\gamma$  is a properly set threshold, the block is considered to be maliciously tampered if  $\rho < \gamma$ . The threshold is determined mathematically or experimentally so as to maximize the probability of tamper detection subject to a given probability of

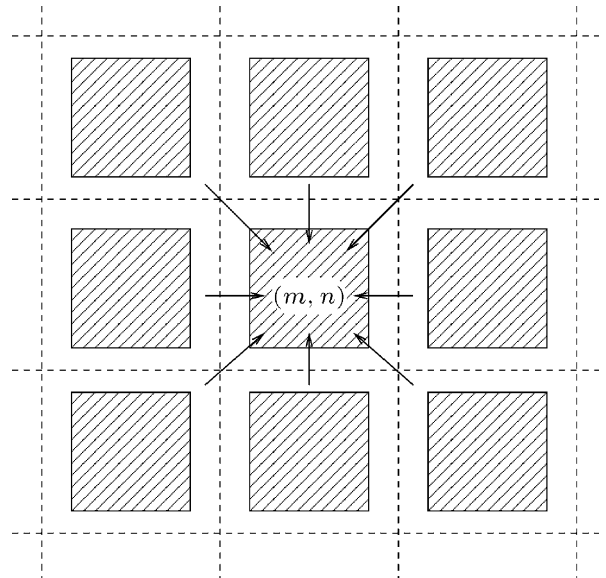


Fig. 4. The inter-block relationship in the pinned sine transform.

false alarm. When a mismatch is observed, there may be confusion in identifying the tampered block between the two paired macro-blocks. Assume that the tampering is localized and manipulation of one sub-block disturbs the DCT polarity information of the whole macro-block, we decide some sub-block in  $\mathbf{X}_\mu$  was tampered, if the watermark extracted from  $\mathbf{X}_\nu$  does not authenticate this sub-block, while all of the sub-blocks of  $\mathbf{X}_\nu$  cannot be authenticated by watermarks extracted from  $\mathbf{X}_\mu$ .

If some parts of the watermarked image were detected to be removed or destroyed, they would be automatically restored using the method described in Section 2.1. The macro-block containing tampered

blocks is viewed as an irregularly sampled signal with lost samples on the locations of the tampered blocks. The tampered blocks are then restored using the following algorithm:

**Algorithm 3.** Restoration of tampered blocks.

$$\begin{aligned} \text{Init } \mathbf{X}^{(0)} &= \mathcal{P}_0 \mathbf{X}, \\ \mathbf{X}^{(i+1)} &= \mathcal{P} \mathbf{X}^{(i)}, \end{aligned}$$

where the projection operator  $\mathcal{P}$  is as that defined in Eq. (4), and  $\mathcal{P}_0$  is spatial domain projection defined by

$$\mathcal{P}_0 \mathbf{y} = \begin{cases} 0 & \text{if } n \in \Delta, \\ y(n) & \text{if } n \notin \Delta, \end{cases} \quad (14)$$

with  $\Delta$  denoting the tampered sub-blocks detected in the authentication stage. The polarity informa-

tion,  $\Gamma(u, v)$ , has been extracted from the paired macro-block of the tampered macro-block.

#### 2.4. Minimization of the probability of false alarm

In practice, the DCT domain polarity information extracted from the macro-blocks may change after its contained sub-blocks are watermarked. This leads to false alarm during image authentication. False alarms also occur when images are subject to lossy transcoding or other common image processing operations which are viewed legitimate since they preserve the content of the images. Therefore, two compensative schemes are developed to minimize the probability of false alarm. Firstly, a detection process is performed immediately after the embedding. If a false alarm occurs on any block, an

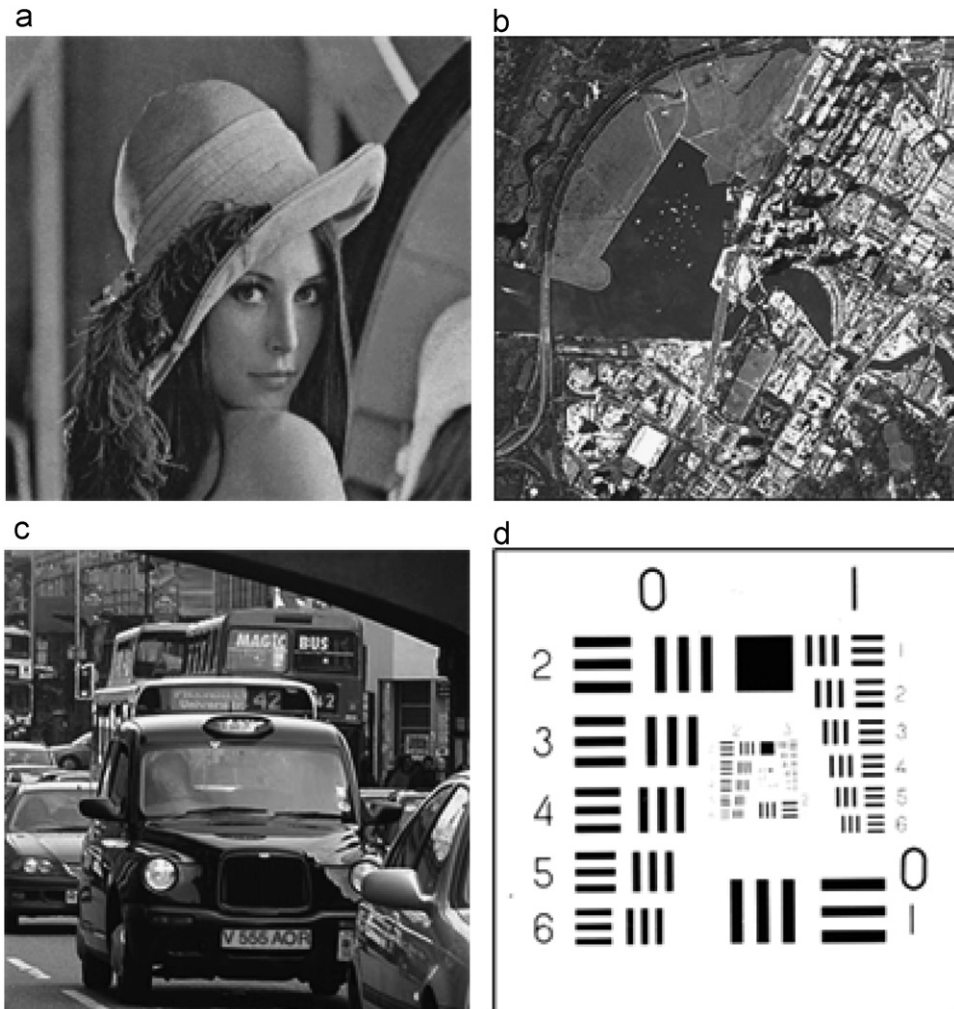


Fig. 5. The testing images: (a) Lena; (b) Singapore; (c) Traffic; (d) Chart.

iterative embedding procedure is performed on those blocks based on the newly extracted polarity information.

The second solution is to be executed in the authentication process. When the system detects “wrong” blocks, a verification process is activated before outputting the final tampered block alarm. The verification process is a variation of the restoration process:

**Algorithm 4.** Verification of tampered blocks.

$$\text{Init } \mathbf{X}^{(0)} = \mathbf{X},$$

$$\mathbf{X}^{(i+1)} = \mathcal{P}\mathbf{X}^{(i)}.$$

This iteration procedure is performed on the detected “wrong” block for  $\eta$  times and the restored

result is compared with the input test image. In a false alarm case, since the initial guess of the restoration is supposed to be the true content, the restoration process hardly changes the image. It is judged by the system as a false alarm if the PSNR of

Table 1  
Comparison of the PSNRs (dB) of the watermarked images

Image	Proposed method	Fridrich 1 (1 LSB)	Fridrich 1 (2 LSBs)	Fridrich 2 (differential coding)
Lena	36.73	50.43	43.78	33.10
Singapore	35.91	51.12	43.80	32.21
Traffic	35.19	51.06	43.91	31.57
Chart	38.03	54.02	46.65	29.23

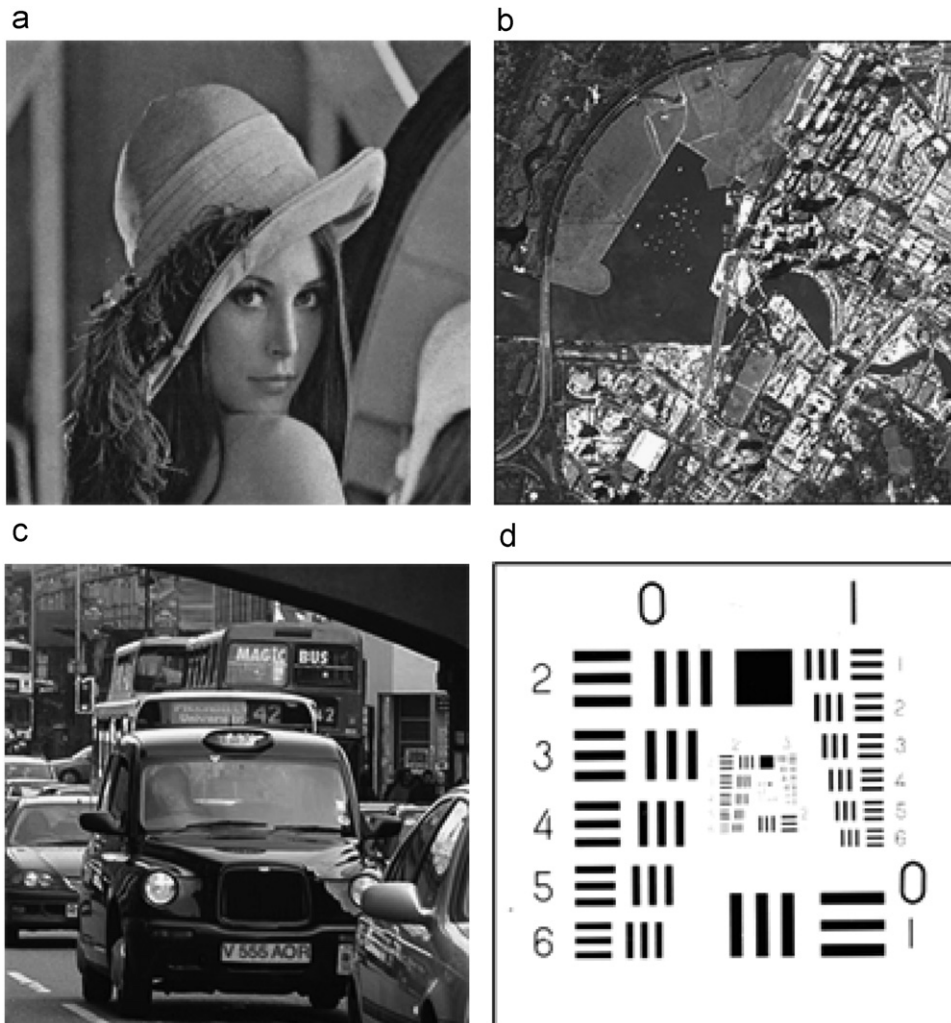


Fig. 6. The watermarked images: (a) Lena; (b) Singapore; (c) Traffic; (d) Chart.



Fig. 7. Simulation results: (a) the tampered image; (b) the authentication result; (c) the restored image after 5 iterations; (d) the restored image after 20 iterations.

the restored image compared to the input test image is larger than a threshold  $T_p$ . For the consideration of computational complexity of the system,  $\eta$  should be kept the minimum number which provides discrimination between tampered blocks and true blocks. In our experiments, we found  $\eta = 3$  is enough for this purpose.

### 2.5. Security against malicious attacks

The most important security issue for authentication watermarking system is the block replacement attacks [8]. The attacker would replace a watermarked block by another block which is embedded with the same watermark pattern with the target block. The system would fail to detect this tampering since the extracted watermark is unaffected. For such attacks to succeed, the assumption is that the attacker possesses some knowledge about the

watermark patterns. In our proposed system, it is possible that the attacker would change the content of a sub-block while keeping the polarities of the pinned sine transform coefficients unchanged.

To solve this problem, we can pseudo-randomly select the coefficients for embedding so that the possibility that the attackers succeed in identifying the target coefficients would be minimized. Another solution to avoid such attacks is to introduce inter-relationship between the watermarked blocks. In our scheme, we exploit the intrinsic inter-block dependence in PST to detect the above counterfeiting attacks. The ‘‘PST style’’ encoding<sup>4</sup> introduces an inter-block relationship to the pinned sine transformed images as shown in Fig. 4. Thus, block replacement counterfeiting attacks can be exposed

<sup>4</sup>Refer to Eqs. (15)–(24).

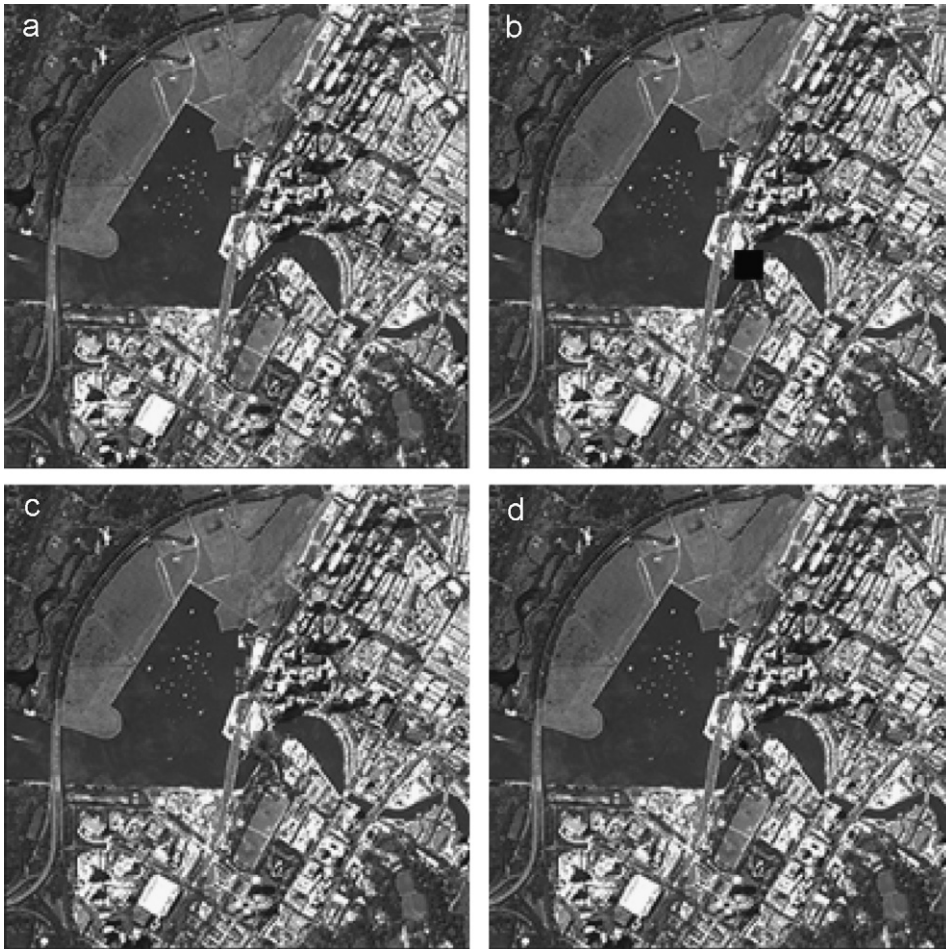


Fig. 8. Simulation results: (a) the tampered image; (b) the authentication result; (c) the restored image after 5 iterations; (d) the restored image after 20 iterations.

by this encoding style since the counterfeiting of one block affects all the surrounding blocks. As a tradeoff, the above mentioned inter-block relationship may result in decreased localization accuracy, that is, the system would fail to localize the tampering in a single sub-block. Our solution is to use Algorithm 4 to verify any reported “false block” and minimize the probability of false alarm.

### 3. Simulation results and performance analysis

In this section, the performance of the proposed method is evaluated. In our experiment, the parameters were set as follows:  $n = 8$ ,  $m = 3$ ,  $\ell = 6$ ,  $\lambda = 10$ ,  $\gamma = 0.5$ ,  $\eta = 3$ , and  $T_v = 20$  dB.

Comparison with existing techniques has also been performed. The  $256 \times 256$  gray-scale images as shown in Fig. 5 were used to test our method.

#### 3.1. The quality of watermarked images

The watermarked images are given in Fig. 6 and perceived identical to the original images. Table 1 shows the comparison of PSNRs of the watermarked images with those of Fridrich methods [6] (LSB embedding methods using 1 bit and 2 bits, respectively, and the differential coding method). As expected, the proposed method introduced relatively more degradation than the LSB methods; however, the quality is far better than the differential coding method.

#### 3.2. Authentication and restoration

The watermarked image Fig. 6(a) was modified as shown in Fig. 7(a): the flower on Lena’s hat was removed. As shown in Fig. 7(b), this modification

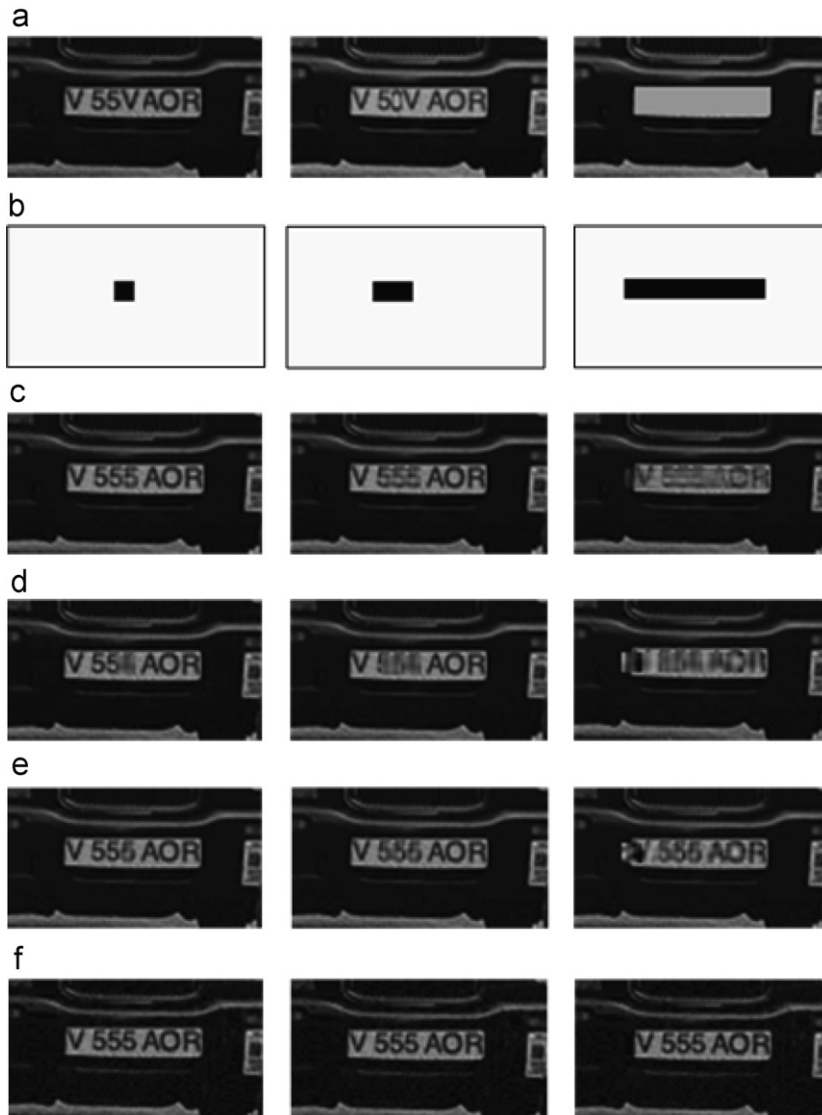


Fig. 9. Simulation results: (a) the tampered images; (b) authentication results; (c) restored images by the proposed method; (d) restored images by Fridrich method 1 (1 LSB); (e) restored images by Fridrich method 1 (2 LSBs); (f) restored images by Fridrich method 2.

Table 2  
Comparison of the PSNRs (dB) of the restored portions in Fig. 9

Attacks	Proposed method	Fridrich 1 (1 LSB)	Fridrich 1 (2 LSBs)	Fridrich 2 (differential coding)
Col. 1	22.80	17.70	21.50	28.82
Col. 2	20.56	17.41	21.01	29.00
Col. 3	15.98	17.23	20.14	28.99

was accurately identified by the authentication scheme. The restoration result after five iterations is shown in Fig. 7(c), which is still not recognizable with PSNR = 13.36 dB. Fig. 7(d) shows the restora-

Table 3  
Probability of false alarm of the authentication process

JPEG compression		Other attacks	
Quality	$P_{FA}$	Attacks	$P_{FA}$
$QF = 90$	0.0000	Gaussian filtering	0.0000
$QF = 70$	0.0000	Unsharpening	0.0000
$QF = 50$	0.0069	Contrast enhancement	0.0000
$QF = 30$	0.0127	Salt & pepper noise	0.0054
$QF = 10$	0.0138	Median filtering	0.0090

tion result after 50 iterations which is significantly more recognizable with PSNR = 23.30 dB. Fig. 8 illustrates another example. The bridges in Fig. 6(b)

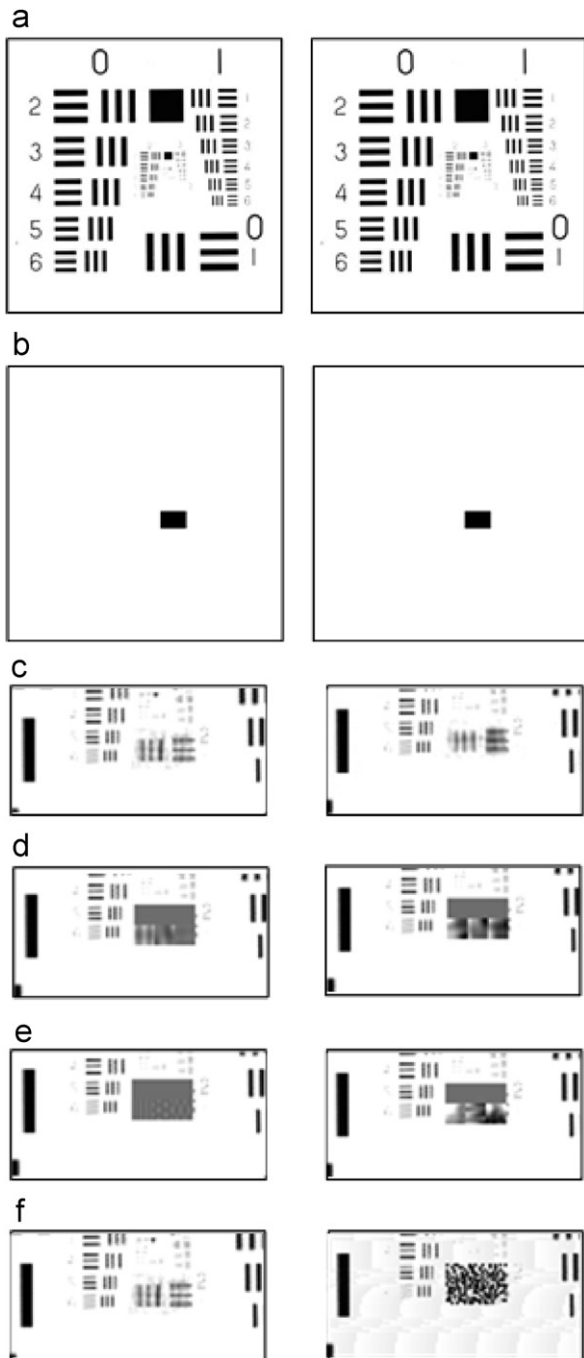


Fig. 10. Simulation results: (a) the tampered images; (b) authentication results; (c) restored images by the proposed method; (d) restored images by Fridrich method 1 (1 LSB); (e) restored images by Fridrich method 1 (2 LSBs); (f) restored images by Fridrich method 2.

were removed and resulted in Fig. 8(a). Fig. 8(b) shows the authentication result, and Figs. 8(c) and (d) show the restoration result after five iterations

Table 4  
Comparison of the PSNRs (dB) of the restored portions in Fig. 10

Attacks	Proposed method	Fridrich 1 (1 LSB)	Fridrich 1 (2 LSBs)	Fridrich 2 (differential coding)
JPEG ( $QF = 90$ )	16.21	6.94	6.83	11.33
JPEG ( $QF = 80$ )	16.19	6.35	6.42	4.66

and after 50 restorations. The PSNR values are 13.84 and 18.13 dB, respectively.

We also compared the performance of our proposed scheme with previous methods. In the watermarked image Fig. 6(c), the licence numbers were maliciously modified. As illustrated in Fig. 9(a), modifications with different area sizes were performed. The authentication and restoration results are shown in the Figs. 9(b) and (c), respectively. We observe that the tampered blocks were accurately detected and the restored results were visually acceptable. Figs. 9(d)–(f) show the restoration results by Fridrich methods for a comparison, and Table 2 shows the PSNRs of the restored portions. We observe that the proposed method outperformed the LSB methods. The differential coding method obtained better restoration results than the proposed method here; however, as shown in Table 1, the visual quality of the watermarked images resulted from differential coding was not satisfactory. We also found that, in our proposed method, the quality of the restored image degrades when the area of the tampered portion becomes larger. This problem can be solved by increasing the size of macro-blocks, although this will also increase the computational overhead.

To test the probabilities of false alarm of the authentication caused by various image processing operations, after modification of contents, the images were further degraded by other acceptable manipulations. Table 3 lists testing results obtained on an image database of 1000 images. It can be observed that our scheme maintains a low probability of  $P_{FA} < 0.01$  for JPEG compression with  $QF \geq 50$  and other processing attacks including the median filtering, which is considered to be very difficult for the successful detection of watermarks.

Fig. 10 shows the restoration results after both content alteration and JPEG compression. Some portion in Fig. 6(d) was removed and the image was

further JPEG compressed. The first column in Fig. 10 illustrates the results after JPEG compression with quality factor of 90%, and the second column illustrates the results after JPEG compression with quality factor of 80%. The LSB embedding methods failed to restore the removed parts. The differential coding method survived the 90% JPEG compression; however, it failed when the quality factor decreased to 80%. The proposed approach achieved acceptable result with PSNR of approximately 16 dB. The PSNRs of the restored parts are listed in Table 4.

A benchmark evaluation of the robustness of restoration using 1000 images was also performed and the results are reported in Table 5. As illustrated in Table 5, for the proposed approach, satisfactory results (e.g., PSNR of approximately 20 dB) are assured, except for the extreme situations of JPEG compression with  $QF = 10$  and median filtering. In all the cases, the proposed method obtained far better results than Fridrich's methods.

### 3.3. Convergence of restoration process

The convergence of Algorithm 4 was investigated on a database of 1000 natural images as shown in Fig. 11. During the evaluation, we set  $n = 8$ ,  $m = 3$

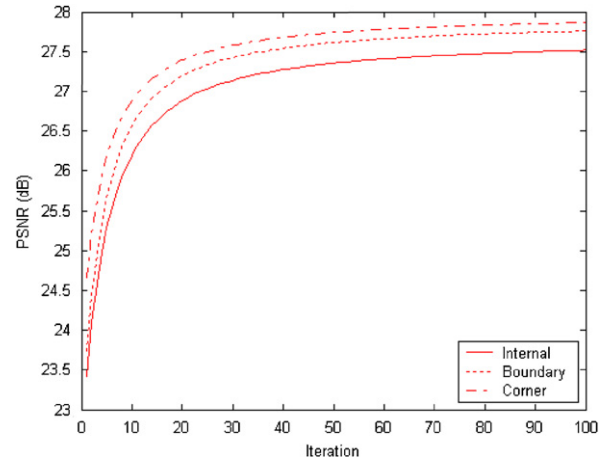


Fig. 11. Convergence of Algorithm 3.

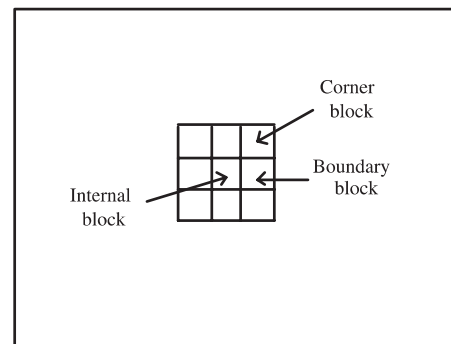


Fig. 12. Different locations of tampered sub-blocks within a macro-block with  $m = 3$ .

Table 5  
Robustness of the self-restoration

	PSNR of restored blocks (dB) <sup>a</sup>			
	Proposed method	Fridrich 1 (1 LSB)	Fridrich 1 (2 LSBs)	Fridrich 2 (differential)
<i>JPEG compression</i>				
<i>Quality</i>				
$QF = 90$	23.94	5.30	5.67	14.4
$QF = 70$	22.75	5.48	5.73	7.49
$QF = 50$	22.56	5.13	5.66	7.42
$QF = 30$	19.27	5.13	5.76	7.42
$QF = 10$	13.65	4.81	5.04	7.95
<i>Other attacks</i>				
<i>Attacks</i>				
Gaussian filtering	22.62	5.99	5.94	8.71
Unsharpening	20.03	5.00	5.17	8.04
Contrast enhancement	22.63	3.54	3.50	8.43
Salt & pepper noise	20.87	19.01	17.17	14.59
Median filtering	12.5	8.05	7.48	8.78

<sup>a</sup>Compared with the original unwatermarked image.

and  $\ell = 6$ . After the images were watermarked as described in Section 2.2, each sub-block was set to be null and restored based on Algorithm 3.

We group the tampered sub-blocks into three types: the blocks inside a macro-block, those on the boundary of a macro-block and those on the four corners of a macro-block as illustrated in Fig. 12. An average result is reported for every group. We notice that the restoration process is insensitive to the locations of the blocks. Moreover, the PSNR of the reconstructed block converged to a satisfactory value ( $> 27$  dB) after approximately 50 iterations.

## 4. Conclusions and future work

In this paper, a semi-fragile watermarking method was proposed for automatic image content authentication and restoration. The problem of restoration of tampered images was expressed as an irregular sampling problem. The tampered image can be reconstructed through iterative projections

onto convex sets. Prior knowledge was hashed into the watermark signal and embedded into the pinned field of PST of the original image. Experimental results showed that accurate authentication and restoration were assured under lossy transcoding and other common image processing operations such as filtering. Our future work will focus on the problem of reconstructing tampered images with larger tampered area with watermarking. The current work can be viewed as a new perspective in unifying the watermarking, authentication and restoration problems into an integrated field that would help in improvements of the tampering detection accuracy and the robust restoration.

### Acknowledgments

The authors appreciate the area editor and the anonymous reviewers for giving valuable comments which have helped to improve the technical content of this paper. The first author would like to thank Dr. Dan He in Centre for Communication Systems Research, University of Surrey, for his useful discussions and suggestions.

### Appendix A. The pinned sine transform

An image  $\mathbf{X}$  is partitioned into non-overlapping blocks of size  $N \times N$  as shown in Fig. 13. Let us consider a typical block  $\mathbf{x}_{m,n}$ , where  $m$  and  $n$  are the coordinate numbers of this block, we define its corner response as

$$\mathbf{c}_{m,n} = \{c_{11}, c_{1N}, c_{N1}, c_{NN}\} \quad (15)$$

and its boundary response as

$$\mathbf{b}_{m,n} = \{\mathbf{b}_{1x}, \mathbf{b}_{Nx}, \mathbf{b}_{y1}, \mathbf{b}_{yN}\} \quad (16)$$

as illustrated in Fig. 13. The corner response is obtained using the corner function:

$$\mathbf{c}_{m,n} = \mathcal{C}[\mathbf{x}_{u,v} : m-1 \leq u \leq m+1, n-1 \leq v \leq n+1]. \quad (17)$$

More specifically, the corner function is defined as follows:

$$c_{11} = \frac{\mathbf{x}_{m,n}(1,1) + \mathbf{x}_{m-1,n-1}(N,N) + \mathbf{x}_{m-1,n}(N,1) + \mathbf{x}_{m,n-1}(1,N)}{4},$$

$$c_{1N} = \frac{\mathbf{x}_{m,n}(1,N) + \mathbf{x}_{m-1,n}(N,N) + \mathbf{x}_{m-1,n+1}(N,1) + \mathbf{x}_{m,n+1}(1,1)}{4},$$

$$c_{N1} = \frac{\mathbf{x}_{m,n}(N,N) + \mathbf{x}_{m,n-1}(N,N) + \mathbf{x}_{m+1,n-1}(1,N) + \mathbf{x}_{m+1,n}(1,1)}{4},$$

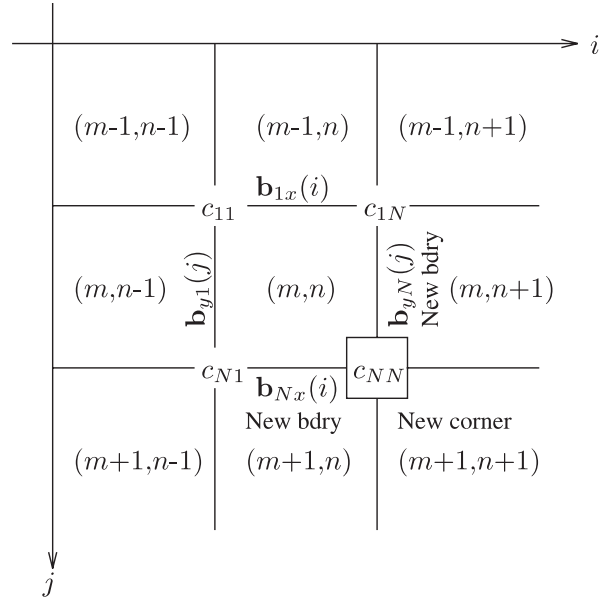


Fig. 13. The dual-field decomposition in PST for a typical block.

$$c_{NN} = \frac{\mathbf{x}_{m,n}(N,N) + \mathbf{x}_{m,n+1}(N,1) + \mathbf{x}_{m+1,n}(1,N) + \mathbf{x}_{m+1,n+1}(1,1)}{4}, \quad (18)$$

and the boundary response is defined by the boundary function

$$\mathbf{b}_{m,n} = \mathcal{B}[\mathbf{x}_{u,v} : m-1 \leq u \leq m+1, n-1 \leq v \leq n+1] \quad (19)$$

which is further defined as

$$\mathbf{b}_{1x}(i) = \frac{\mathbf{x}_{m,n}(1,i) + \mathbf{x}_{m-1,n}(N,i)}{2},$$

$$\mathbf{b}_{Nx}(i) = \frac{\mathbf{x}_{m,n}(N,i) + \mathbf{x}_{m+1,n}(1,i)}{2},$$

$$\mathbf{b}_{y1}(j) = \frac{\mathbf{x}_{m,n}(j,1) + \mathbf{x}_{m,n-1}(j,N)}{2},$$

$$\mathbf{b}_{yN}(j) = \frac{\mathbf{x}_{m,n}(j,N) + \mathbf{x}_{m,n+1}(j,1)}{2}. \quad (20)$$

As we can see from Eqs. (17)–(20), the processing of one block should involve all the blocks surrounding it, and we can observe in Fig. 13 that in a sequential processing of blocks, only one new corner  $c_{NN}$  and two new boundaries  $\mathbf{b}_{Nx}$  and  $\mathbf{b}_{yN}$  are required to be computed for a new input block.

The boundary field of  $\mathbf{x}_{m,n}$  is achieved by the pinning function [14]:

$$\mathbf{x}_{m,n}^b = \mathcal{P}\mathcal{I}\mathcal{N}[\mathbf{c}_{m,n}, \mathbf{b}_{m,n}]. \quad (21)$$

Corresponding to the above general form, the specific form of the pinning function is defined as follows:

$$\begin{aligned} \mathbf{x}_{m,n}^b(i,j) = & \mathbf{x}_{m,n}(1,1) + \frac{(c_{1N} - c_{11})(i - \frac{1}{2})}{N} \\ & + \frac{(c_{N1} - c_{11})(j - \frac{1}{2})}{N} \\ & + \frac{(c_{11} + c_{NN} - c_{N1} - c_{1N})(i - \frac{1}{2})(j - \frac{1}{2})}{N^2} \\ & + \frac{\mathbf{g}_x(i) + (\mathbf{h}_x(i) - \mathbf{g}_x(i))j - \frac{1}{2}}{N} \\ & + \frac{\mathbf{g}_y(j) + (\mathbf{h}_y(j) - \mathbf{g}_y(j))i - \frac{1}{2}}{N}, \end{aligned} \quad (22)$$

where

$$\begin{aligned} \mathbf{g}_x(i) &= \mathbf{b}_{Nx}(i) - \left( c_{N1} + \frac{c_{NN} - c_{N1}}{N} \left( i - \frac{1}{2} \right) \right), \\ \mathbf{h}_x(i) &= \mathbf{b}_{1x}(i) - \left( c_{11} + \frac{c_{1N} - c_{11}}{N} \left( i - \frac{1}{2} \right) \right), \\ \mathbf{g}_y(j) &= \mathbf{b}_{yN}(j) - \left( c_{1N} + \frac{c_{NN} - c_{1N}}{N} \left( j - \frac{1}{2} \right) \right), \\ \mathbf{h}_y(j) &= \mathbf{b}_{y1}(j) - \left( c_{11} + \frac{c_{N1} - c_{11}}{N} \left( j - \frac{1}{2} \right) \right) \end{aligned} \quad (23)$$

are the pinned boundaries. The pinned field  $\mathbf{x}_{m,n}^p$  is then given by

$$\mathbf{x}_{m,n}^p = \mathbf{x}_{m,n} - \mathbf{x}_{m,n}^b. \quad (24)$$

Next, we perform a sine transform to this pinned field block as follows:

$$\mathbf{x}_{m,n}^{p(s)} = \mathbf{S}_N \mathbf{x}_{m,n}^p \mathbf{S}_N^T, \quad (25)$$

with  $\mathbf{S}_N$  defined as in Eq. (10).

## References

- [1] F. Bao, R.H. Deng, B.C. Ooi, Y. Yang, Tailored reversible watermarking schemes for authentication of electronic clinical atlas, *IEEE Trans. Inform. Technology in Biomedicine* 9 (4) (December 2005) 554–563.
- [2] P.S.L.M. Barreto, H.Y. Kim, V. Rijmen, Toward secure public-key blockwise fragile authentication watermarking, *IEE Proc. Vis. Image Signal Process.* 149 (2) (April 2002) 57–62.
- [3] M.U. Celik, G. Sharma, E. Saber, A.M. Tekalp, Hierarchical watermarking for secure image authentication with localization, *IEEE Trans. Image Process.* 11 (6) (June 2002) 585–595.
- [4] M.U. Celik, G. Sharma, A.M. Tekalp, Lossless watermarking for image authentication: a new framework and an implementation, *IEEE Trans. Image Process.* 15 (4) (April 2006) 1042–1048.
- [5] I.J. Cox, M.L. Miller, J.A. Bloom, *Digital Watermarking*, Morgan Kaufman Publishers, San Francisco, CA, USA, 2001.
- [6] J. Fridrich, M. Goljan, Images with self-correcting capabilities, in: *IEEE International Conference on Image Processing*, Kobe, Japan, October 1999, pp. 792–796.
- [7] A.T.S. Ho, X. Zhu, Y.L. Guan, Image content authentication using pinned sine transform, *EURASIP J. Appl. Signal Process.* 2004 (14) (October 2004) 2174–2184 (Special Issue on Multimedia Security and Rights Management).
- [8] M. Holliman, N. Memon, Counterfeiting attacks on oblivious block-wise independent invisible watermarking schemes, *IEEE Trans. Image Process.* 9 (3) (March 2000) 432–441.
- [9] A.K. Jain, *Fundamentals of Digital Image Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1989.
- [10] D. Kundur, D. Hatzinakos, Digital watermarking for telltale tamper proofing and authentication, *Proc. IEEE* 87 (7) (July 1999) 1167–1180.
- [11] C.-S. Lu, H.-Y.M. Liao, Structural digital signature for image authentication: an incidental distortion resistant scheme, *IEEE Trans. Multimedia* 5 (2) (June 2003) 161–173.
- [12] P. Marziliano, M. Vetterli, Irregular sampling in approximation subspaces, *SampTA99*, Loen, Norway, August 1999.
- [13] A.Z. Meiri, E. Yudilevich, A pinned sine transform image coder, *IEEE Trans. Commun.* COM-29 (December 1981) 1728–1753.
- [14] P.W. Wong, N. Memon, Secret and public key image watermarking schemes for image authentication and ownership verification, *IEEE Trans. Image Process.* 10 (10) (October 2001) 1593–1601.
- [15] D.C. Youla, Mathematical theory of image restoration by the method of convex projections, in: H. Stark (Ed.), *Image Recovery: Theory and Application*, Academic Press, Florida, 1987, pp. 29–77.
- [16] Y. Zhao, P. Campisi, D. Kundur, Dual domain watermarking for authentication and compression of cultural heritage images, *IEEE Trans. Image Process.* 13 (3) (March 2004) 430–448.