Signal Processing 92 (2012) 1525–1535

Contents lists available at SciVerse ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

A novel multi-group exploiting modification direction method based on switch map

Xing-Tian Wang [a], Chin-Chen Chang [b,c,*], Chia-Chun Lin [d], Ming-Chu Li [a]

[a] Department of Software, Dalian University of Technology, DaLian, China
[b] Department of Information Engineering and Computer Science, Feng Chia University, Taichung City 407, Taiwan
[c] Department of Biomedical Imaging and Radiological Science, Chinese Medical University, Taichung City 404, Taiwan
[d] Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi 621, Taiwan

ARTICLE INFO

Article history:
Received 12 August 2011
Received in revised form
23 November 2011
Accepted 14 December 2011
Available online 29 December 2011

Keywords:
Data hiding
Exploiting modification direction
Steganography
Switch map

ABSTRACT

In this paper, we proposed a novel adjustable data hiding method. Our proposed method, which is based on exploiting modification direction (EMD) method, is called the "multi-group exploiting modification direction" method. In the multi-group method, we combine several pixel-groups to embed secret data according to the constructed switch map to simply avoid the conversion redundancy of the EMD method and the spatial redundancy of the section-wise strategy, as well as to efficiently increase the probability of achieving larger embedding capacity. According to the experimental results and mathematical justification, we verified that the multi-group method can achieve higher embedding payload and better visual quality of the image than the EMD method and section-wise strategy for simulated and real secret data.

1. Introduction

In recent years, transmitting messages or digital data over the Internet has become quite common and convenient. Then, to ensure the confidentiality of transmitted information between the sender and receiver, the security of information dissemination in an open network is an important issue. Cryptography can encrypt sensitive data before transmission into meaningless data to preserve its content [1–4]. However, meaningless, sensitive data might attract the attention of interceptors and be attacked.

In contrast, data hiding, also known as information hiding, can imperceptibly embed secret data into the meaningful cover media, such as an image, audio, or video [5–7]. Techniques that can embed secret data into an image have attracted the attention of many researchers. Since the secret information embedded into the original image cannot be differentiated from the stego image visually, its presence is not perceptible to potential attackers. However, increasing the embedding payload and maintaining the visual quality simultaneously are competing priorities. As embedding capacity increases, visual quality diminishes, and the reverse is true as well.

Currently, many data hiding methods have been proposed to embed secret data into images. The least significant bit (LSB) replacement [8–11] is a commonly used simple hiding strategy that replaces the least significant bits of cover pixels with the secret bits. However, the attacker can detect the embedded secret bits by statistical analysis [12]. Hence, Mielikainen proposed an LSB matching revisited scheme [13] to embed two secret bits into a pair of cover pixels at one time but modify one of the two pixels. Although the payload of Mielikainen's scheme is the same as that of LSB, it has higher visual quality because of fewer modifications to the cover image.

In 2006, Zhang and Wang [14] proposed a data hiding method exploiting modification direction (EMD), which fully exploits the modification directions of Mielikainen's

* Corresponding author at: Department of Information Engineering and Computer Science, Feng Chia University, Taichung City 407, Taiwan.
E-mail address: alan3c@gmail.com (C.-C. Chang).

0165-1684/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2011.12.013

scheme. Theoretically, the maximal embedding rate of the EMD method is approximately 1.16 bpp. However, realistically, the practical embedding rate cannot achieve this value because of the existence of redundancy caused by converting binary bits into a stream of digits in a $(2n+1)$-ary notational system. Recently, many researchers have proposed various schemes to improve the EMD method [15–24]. Some of these methods utilized the combination of two or more different codes or double-layer embedding to enhance the embedding performance. Some methods improved imperceptibility using some optimization methods. In 2010, Wang et al. [25] proposed an improved section-wise exploiting modification direction method, which combines two pixel-groups to embed the corresponding secret message. This method can improve the image quality while maintaining an embedding capacity similar to that of the EMD method. However, it causes a problem of spatial redundancy.

In this paper, we proposed a switch map algorithm to solve the conversion redundancy of the EMD method and the spatial redundancy of the section-wise strategy. In addition, it increases the probability of achieving larger embedding capacity. Then, we proposed an adjustable multi-group method based on switch map to improve the embedding capacity and image quality of the EMD method and the section-wise strategy. According to the experimental results and the mathematical justification, we can prove that our proposed multi-group method has advantages over the EMD method and the section-wise strategy.

The rest of this paper is organized as follows. The EMD method and section-wise strategy are described briefly in Section 2. In Section 3, the proposed multi-group method is introduced in detail to explain how redundancy is solved and embedding capacity and image quality is improved. The experimental results and relevant discussion are presented in Section 4. Finally, we provide our conclusions in Section 5.

2. Related work

In this section, we will briefly introduce EMD method and section-wise strategy, which improves the visual quality of EMD method.

2.1. The EMD method

A novel steganographic method exploiting modification direction (EMD) was proposed by Zhang and Wang [14] in 2006. In the EMD method, a group of n pixels has $(2n+1)$ possible directions to conceal secret data, i.e., $2n$ ways of modification plus one unchanged case. The authors proved that the $(2n+1)$ directions are mutually different, i.e., each embedding direction is unique, so the embedded secret digits can be extracted accurately.

In the pre-processing procedure, according to a secret key, all the pixels of the cover image are permuted pseudo-randomly and partitioned into a sequence of pixel-groups, each containing n pixels. The pixel-group is denoted as $G=(g_1, g_2, \ldots, g_n)$, where g_i represents the grayscale value of the ith pixel in the pixel-group. On the other hand, the binary secret bits are converted into a sequence of secret digits in a $(2n+1)$-ary notational system.

The extraction function f can be calculated as a weighted sum modulo $(2n+1)$:

$$f(g_1,g_2,\ldots,g_n)=\left[\sum_{i=1}^{n}(g_i\times i)\right] \bmod (2n+1). \tag{1}$$

First, in the embedding stage, we use the extraction function to get the value f of a pixel-group. Then, we can use Eq. (2) to get the value d for a secret digit s:

$$d=(s-f) \bmod (2n+1). \tag{2}$$

Finally, we can get the modified group G' using Eqs. (3)–(5):

1. If $s=f$, $G'=(g'_1,g'_2,\ldots,g'_n)=(g_1,g_2,\ldots,g_n)$. (3)

2. If $s\neq f$ and $d\leq n$, $g'_i=\begin{cases} g_i, & \text{if } i\neq d \\ g_i+1, & \text{if } i=d \end{cases}$ for $i=1,2,\ldots,n$. (4)

3. If $s\neq f$ and $d>n$, $g'_i=\begin{cases} g_i, & \text{if } i\neq 2n+1-d \\ g_i-1, & \text{if } i=2n+1-d \end{cases}$ for $i=1,2,\ldots,n$. (5)

If the modified pixel value is less than 0 or more than 255, the pixel must be in an underflow or overflow situation. To solve this problem, the corresponding original pixel must be increased or decreased by 1 before embedding. After adjusting the saturated pixel, the secret digit will be embedded into the adjusted pixel-group again.

After the embedding stage, the permuted stego pixel stream embedded the secret digits are rearranged to the sequence of the original image with the same secret key to obtain the stego image.

The theoretical maximal embedding rate of the EMD method, $R=((\log_2(2n+1))/n)$, is 1.16 bpp (bit per pixel) for the case $n=2$. Since, at most, only one pixel of each pixel-group is increased or decreased by 1, the EMD method can provide high quality images.

2.2. Section-wise strategy

Wang et al. proposed an improved EMD method in 2010 using the section-wise strategy [25]. The method provides better visual quality while maintaining an embedding capacity that is similar to that of the EMD method.

Different from the EMD method, the section-wise strategy combines two pixel-groups to indicate the available modification directions. In the embedding procedure, the authors first utilized chaotic sequence to permute the pixel order of the cover image. Then, they provided a 2-level section, i.e., using the selector pointer (SP) and the descriptor pointer (DP) to assign to a selective group and a descriptive group, respectively. The value of SP or DP varies from 0 to $2n$, where n represents the number of pixels in the selective group or the descriptive group. Once the range of SP and DP are decided, a table can be built in which SP and DP point the rows and columns of

	DP=0	DP=1	DP=2	DP=3	DP=4
SP=0	00000	00001	00010	00011	00100
SP=1	00101	00110	00111	01000	01001
SP=2	01010	01011	01100	01101	01110
SP=3	01111	10000	10001	10010	10011
SP=4	10100	10101	10110	10111	11000
SP=5	11001	11010	11011	11100	11101
SP=6	11110	11111	Empty	Empty	Empty

Fig. 1. 5×7 selector and descriptor table.

the table, respectively. The values of SP and DP, which are the projections mapped from the to-be-embedded secret stream in the constructed table, are the corresponding modification directions of the selective group and the descriptive group, respectively. The data hider embeds the secret stream into the selective group and the descriptive group by applying the EMD embedding procedure according to SP and DP. After all secret data are embedded into the cover image, all stego pixels are reshaped by the same chaotic sequence and permutation key to obtain the stego image.

For example, the selective group and the descriptive group contain three and two pixels, respectively. Therefore, the range of SP is [0,6] and the range of DP is [0,4]. We can arrange the table as shown in Fig. 1.

The embedding rate of the section-wise strategy can be calculated according to

$$R = \frac{\lfloor \log_2((2n_{SP}+1) \times (2n_{DP}+1)) \rfloor}{n_{SP}+n_{DP}},$$

and $n_{SP}, n_{DP} \geq 2$, where n_{SP} and n_{DP} represent the number of pixels in the selective group and in the descriptive group, respectively. Therefore, its maximal embedding rate is 1.00 bpp with $n_{SP}+n_{DP}=4$ or 5. Moreover, the author proved that the section-wise strategy can provide higher visual quality of the stego image than the EMD method.

3. Proposed method

In this section, to solve the conversion redundancy of the EMD method and the spatial redundancy of section-wise strategy, first, we will introduce a switch map algorithm. Then a novel adjustable embedding method, i.e., multi-group based on switch map, is proposed to improve the embedding capacity and image quality of the EMD method and the section-wise strategy.

3.1. The analysis of redundancy

In the EMD method, the embedding rate is represented as $R=((\log_2(2n+1))/n)$, where n is the pixel number in each pixel-group. Therefore, the maximal embedding rate of EMD is 1.16 bpp when $n=2$. However, this result is theoretical and not practical because of the redundancy caused by the conversion from the binary stream into a sequence of $(2n+1)$-ary digits. The binary stream is segmented into many pieces with L bits, and the K digits in the $(2n+1)$-ary notational system represent the decimal value of each secret piece, where:

$$L = \lfloor K \times \log_2(2n+1) \rfloor. \quad (6)$$

For example, the original binary sequence $(1001\ 1011\ 1101)_2$ can be expressed as $(14\ 21\ 23)_5$ in the 5-ary notational system where $L=4$ and $K=2$. Nevertheless, the same binary sequence can be expressed as $(34433)_5$, which is converted from the entire original binary sequence without segmentation. According to the above example, we can find that the number of digits in former is one more than that in latter. Therefore, we can use Eq. (7) to represent the practical embedding rate of EMD:

$$R_{EMD} = \frac{\lfloor K \times \log_2(2n+1) \rfloor}{K \times n}. \quad (7)$$

In the EMD method, the authors assumed that redundancy was close to 0 and could be ignored if L and K were large enough. However, the practical length of the secret data is always finite. Moreover, the crucial problem is that adequate storage hardware is required to solve the increasing computing complexity with the growth of L and K. In other words, the conversion redundancy problem of the EMD method is inevitable in practical application.

In the section-wise strategy, the secret binary stream was not converted to a different notational system. Instead, they utilized the combination of two pixel-groups to represent secret bits. However, some positions in the table are vacant and are not used to hide message as shown in Fig. 1; we call this situation "spatial redundancy."

3.2. Switch map

In this subsection, we design a simple and efficient switch map algorithm to eliminate the two styles of redundancy problem while simultaneously increasing embedding capacity. A map of decimal digits to their corresponding binary representations is constructed using the proposed switch map algorithm. In this map, the lengths of binary representations of some decimal digits have been switched. According to the switch map, we can get certain secret digits, and the lengths of their binary presentations may be increased. We will introduce the switch map algorithm as follows.

For a set of given decimal digits [0, X], we convert these digits into their binary representations BRs, with each BR being $L=\lceil \log_2(X+1) \rceil$ bits, to construct a secondary map. If the BR of a decimal digit is less than L bits, the remainder of the most significant bits is replenished with 0 as shown in Fig. 2(a) with $X=6$. However, some digits with BRs that are $\lceil \log_2(X+1) \rceil$ bits but whose values may be more than X cannot be represented in this secondary map, so we call them unused digits UDs, i.e., $X < (UD)_{10} < 2^L$. The digit that is represented by the first $(L-1)$ bits of UD is the element of the map. Therefore, we call this style of digit switchable digit SD because it has two kinds of representations, i.e., L bits and $(L-1)$ bits, in the secondary map. Therefore, we will search the digits that belong to SD based on UD to remove the ambiguity, representing SD by L or $(L-1)$ bits in the binary system and switch SD to $(L-1)$ bits.

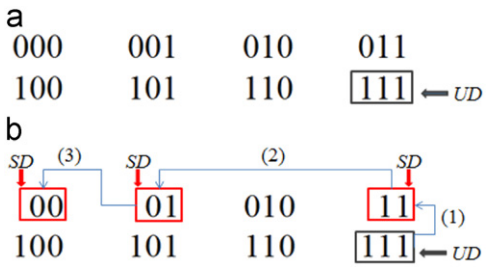

Fig. 2. Process of constructing the switch map. (a) Secondary map and (b) switch map.

We will present an example to explain the switch map algorithm clearly. If $X=(6)_{10}=(110)_2$, $UD=\{(7)_{10}\}=\{(111)_2\}$, and $L=3$, a secondary map of decimal digits [0,6] to their corresponding binary representation can be constructed as shown in Fig. 2(a). According to UD, we extract its first two bits, i.e. $(11)_2=(3)_{10}$, so we switch $(3)_{10}$ to $(11)_2$ as shown by arrow (1) in Fig. 2(b). However, another ambiguous case will occur. Since $(3)_{10}$ has been switched and can be represented by $(11)_2$ in the map, we cannot represent $(011)_2=(3)_{10}$. Then, the first two bits of $(011)_2$ are extracted from it, i.e., $(01)_2=(1)_{10}$. Therefore, we switch $(1)_{10}$ to $(01)_2$ as shown by arrow (2) in Fig. 2(b). In the same way, we can switch $(0)_{10}$ to $(00)_2$ as shown by arrow (3) in Fig. 2(b). Finally, the switch map is constructed successfully. According to the switch map, we can accurately convert the decimal digit and its binary counterpart and vice versa without ambiguity. Simultaneously, we increase the lengths of binary representations of some decimal digits. In the above example, four digits have 3-bit binary counterparts, and three digits have 2-bit binary counterparts.

The switch map can increase the lengths of binary representations of some decimal digits, which can be represented by L bits, so it can be used in data hiding to directly augment the probability of larger embedding capacity. For example, the binary representation of digit $(25)_{10}$ is $(11001)_2$ with the traditional conversion method, while it is $(010101)_2$ with switch map.

In the switch map, for the worst case, the ratio of the digits carrying one more bit than the others to all $(X+1)$ digits can be calculated by Eq. (8):

$$R_{\min}=\frac{1}{A+1},\quad \text{if } X=A,\ \text{where } (\log_2 A)\in N. \tag{8}$$

Therefore, in the procedure of constructing the secondary map, there is no need to fill up the binary presentation of the digit, the value of which is less than X, to L bits if $X=A$, i.e., we represent it by $(L-1)$ bits. In contrast, for the best case, the ratio is:

$$R_{\max}=\frac{B+1-\log_2(B+2)}{B+1},\quad \text{if } X=B,$$
$$\text{where } (\log_2(B+2))\in N \text{ and } B\geq 6. \tag{9}$$

Under the circumstances, i.e., if $X=B$, each digit can be represented by L bits, excluding the ones with values that are equal to (2^i-1), where $(i=0, 1, \ldots, \log_2(B+2)-1)$ and for which their binary presentations are switched to $(L-1)$ bits. The switch map algorithm is described with pseudocode as follows:

```
Definitions:
L = ⌈log2(X+1)⌉   X ≠ 0
u: UD={u1, u2, …, u_{2^L−1−X}}
S_L: The (digit)10 is represented by L bits
Pseudocode:
begin :
    if (i = 1; i ≤ 2^L − 1 − X; i++)
    {
      while ((u_i)2 shift one bit to the right ≠ 0)
      {
          (u_i)10 ∉ S_L;
      }
    }
(0)10 ∉ S_L;
end :
```

3.3. Multi-group method

In this subsection, the proposed adjustable data hiding method based on switch map will be introduced. In this method, we combine some pixel-groups to embed secret data, so we can freely adjust the pixel number of each pixel-group to achieve many combinations of payload and visual quality of image for different actual applications.

Before embedding data, permute all cover pixels pseudo-randomly according to a secret key and divide them into a series of pixel-pieces PPs, each of which contains $\sum_{i=1}^{k}(2n_i+1)$ pixels, i.e., $PP=(g_1,g_2,\ldots, g_{\sum_{i=1}^{k}(2n_i+1)})$ where g_i represents the gray value of the ith pixel in the pixel-piece. Then, each pixel-piece is separated into k pixel-groups PGs, each of which contains n_i pixels, i.e., $PP=(PG_1,PG_2,\ldots,PG_k)$ and $PG_i=(g_{T_i+1}, g_{T_i+2},\ldots,g_{T_i+n_i})$, where $T_i=\sum_{j=1}^{k} n_j$ and $T_1=0$.

After the pre-processing procedure, a switch map is constructed with $X=\Pi_{i=1}^{k}(2n_i+1)-1$. Then $L=\left\lfloor \log_2 \Pi_{i=1}^{k}(2n_i+1)\right\rfloor$ secret bits sb_L are extracted from secret data, and, if they can be presented by the constructed switch map, we denote decimal digit $S=(sb_L)_{10}$, otherwise the first $(L-1)$ bits of sb_L must be represented by the switch map, so $S=(sb_{L-1})_{10}$. The decimal digit S can be decomposed into k secret digits, i.e. $S=(s_1,s_2,\ldots,s_k)$, according to the decomposition algorithm described with pseudocode as follows

```
S=(sb_L)10 or (sb_{L−1})10;
i=0;
while (i < k−1)
{
   s_{k−i} = ⌊ S / ∏_{j=1}^{k−i−1}(2n_j+1) ⌋;
   S = S mod ∏_{j=1}^{k−i−1}(2n_j+1);
   i++;
}
   s_1 = S;
```

Then, we embed each secret digit s_i into each corresponding pixel-group PG_i according to the embedding method of EMD as Eqs. (1)–(5), respectively. After embedding all secret data, all stego pixels are reshaped to the original size and sequence of the cover image with the same secret key to obtain the stego image.

The entire embedding algorithm of the proposed method is summarized as follows:

Step 1: Set parameters, k and n_i.
Step 2: Construct a switch map according to k and n_i.
Step 3: All cover pixels are pseudo-randomly permuted with a secret key and the permuted pixels are divided into a series of pixel-pieces PPs.
Step 4: Each PP is divided into k pixel-groups PGs, i.e., $PP=(PG_1,PG_2, \ldots, PG_k)$.
Step 5: Extract L or $(L-1)$ secret bits, i.e., sb_L or sb_{L-1}, from secret data based on the switch map, then, denote the decimal digit $S=(sb_L)_{10}$ or $(sb_{L-1})_{10}$ and decompose it into k secret digits, i.e., $S=(s_1,s_2, \ldots, s_k)$, according to the decomposition algorithm.
Step 6: Calculate the value f_i of PG_i by Eq. (1).
Step 7: Perform the EMD embedding procedure over PG_i according to s_i and f_i.
Step 8: Read the next PG and s, go to Step 6.
Step 9: Read the next PP and then go to Step 4 until all secrete data are embedded into the cover image.
Step 10: Reshape all stego pixels to the original size and sequence of the cover image with the same secret key to get the stego image.

Fig. 3 displays the flowchart of embedding procedure of our proposed multi-group method.

Next, we will use an example to demonstrate the workflow of multi-group. Assume $k=2$, $n_1=n_2=2$, the pixel-piece is $PP=(180, 181, 181, 180)$, and the binary secret data are (111100). Therefore, $PG_1=(180, 181)$, $PG_2=(181, 180)$, and a switch map ($X=5\times5-1=24$) can be constructed as shown in Fig. 4. The extracted $L=5$ secret bits from secret data, $sb_L=(11110)_2$, cannot be represented by the switch map, but $sb_{L-1}=(1111)_2$ can be represented. According to the decomposition algorithm, two secret digits, $s_1=0$ and $s_2=3$, can be decomposed from sb_{L-1}, and they are embedded into PG_1 and PG_2, respectively. With the embedding method of EMD, the two stego pixel-groups are $PG'_1=(180,180)$ and $PG'_2=(181,181)$.

In multi-group, if $k=1$, it is easier and more intuitive to observe the effect that our method improves EMD. To compare reasonably, since each pixel-group is independent in multi-group due to $k=1$, we also set each pixel-group of EMD as independent, i.e. $K=1$, as described in Section 3.1. For the EMD method, if a pixel-group consists of three pixels, i.e., $n=3$, $L=2$ according to Eq. (6). This means that the binary representations of available modification directions are (00, 01, 10, 11), but the other modification directions (100, 101, 110) cannot be used to embed secret data, so the conversion redundancy problem occurred. On the other hand, in multi-group, a switch map can exploit all seven directions, i.e. $X=6$. Therefore, these directions can be represented as

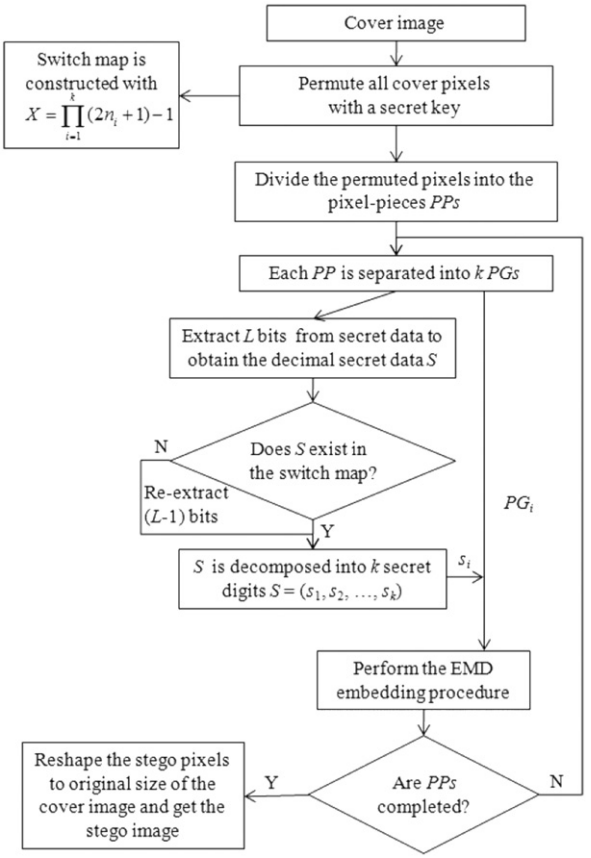

Fig. 3. Flowchart of data embedding procedure.

0000	0001	00010	0011
00100	00101	0110	0111
01000	01001	01010	01011
1100	1101	1110	1111
10000	10001	10010	10011
10100	10101	10110	10111
11000			

Fig. 4. Switch map with $X=24$.

(00, 01, 010, 11, 100, 101, 110) in binary system, as shown in Fig. 2(b), according to the switch map algorithm. In the comparison, our proposed method exploits all modification directions to avoid redundancy and increases the probability of achieving larger embedding capacity.

For example, if the secret bits are (110100), they can be converted as (3, 1, 0) in the 7-ary notational system in EMD. Hence, the three digits will be embedded into three pixel-groups. Nevertheless, the same binary stream can be converted as (6, 4) in the 7-ary notational system with

the switch map and will be embedded into two pixel-groups. Therefore, the one pixel-group that was saved in the latter can be used to carry additional secret digits.

In the case of $k=2$, our multi-group method can be transformed into a table to improve the section-wise strategy. For example, if the ranges of SP and DP are [0,4] in the section-wise strategy, a 5×5 selector and descriptor table can be built with some unused and vacant positions as shown in Fig. 5. In contrast, our method based on switch map can build a full and optimized table as shown in Fig. 6, where Tc (table column) and Tr (table row) are generated according to the pixel numbers of PG_1 and PG_2, respectively. Obviously, the multi-group method has no spatial redundancy problem, and it also can increase the probability of achieving larger embedding capacity.

Data extraction of the multi-group method is simply a calculative process. In extraction, the pre-processing procedure is the same as the embedding procedure. First, all cover pixels are permuted pseudo-randomly with the same key and divided into a series of pixel-pieces; then, each pixel-piece is separated into k pixel-groups, i.e. $PP=(PG_1,PG_2,\ldots,PG_k)$, each of which contains n_i pixels. The same switch map can be constructed with $X=\Pi_{i=1}^{k}(2n_i+1)-1$. According to Eq. (1), the vector of value f, $VF=(f_1, f_2, \ldots, f_k)$, can be obtained by calculating f_i from PG_i. A decimal secret digit SD can be calculated from VF according to Eq. (10):

$$SD=\sum_{i=1}^{k}\left(f_i\times\prod_{j=1}^{i-1}(2n_j+1)\right),$$

$$\text{where}\ \prod_{j=1}^{0}(2n_j+1)=1. \tag{10}$$

With the constructed switch map, SD will be mapped as a binary stream with $L=\left\lfloor \log_2\Pi_{i=1}^{k}(2n_i+1)\right\rfloor$ bits or $(L-1)$ bits. The binary stream is the extracted secret data.

	DP=0	DP=1	DP=2	DP=3	DP=4
SP=0	0000	0001	0010	0011	0100
SP=1	0101	0110	0111	1000	1001
SP=2	1010	1011	1100	1101	1110
SP=3	1111	Empty	Empty	Empty	Empty
SP=4	Empty	Empty	Empty	Empty	Empty

Fig. 5. 5×5 selector and descriptor table built by the section-wise strategy.

	Tc=0	Tc=1	Tc=2	Tc=3	Tc=4
Tr=0	0000	0001	00010	0011	00100
Tr=1	00101	0110	0111	01000	01001
Tr=2	01010	01011	1100	1101	1110
Tr=3	1111	10000	10001	10010	10011
Tr=4	10100	10101	10110	10111	11000

Fig. 6. 5×5 full and optimized table built by the multi-group method.

The extraction algorithm of the multi-group method is concluded as follows:

Step 1: Construct a switch map according to given k and n_i.
Step 2: All pixels of the stego image are pseudo-randomly permuted with the given secret key and the permuted pixels are divided into a series of pixel-pieces PPs.
Step 3: Each PP is divided into k pixel-groups PGs, i.e., $PP=(PG_1,PG_2, \ldots, PG_k)$.
Step 4: Obtain $VF=(f_1, f_2, \ldots, f_k)$ by calculating f_i of PG_i according to Eq. (1).
Step 5: Calculate a decimal secret digit SD from VF using Eq. (10).
Step 6: Convert SD into L or $(L-1)$ secret bits with the switch map.
Step 7: Read the next PP and then go to Step 3 until all secrete data are extracted from the stego image.

4. Experimental results

In this section, some experimental results and relevant discussions are provided to demonstrate the embedding performance of our proposed method. Six standard grayscale images were selected as the test cover images with resolutions of 512×512 pixels as shown in Fig. 7. Two experiments were designed to compare our proposed method with the EMD method and the section-wise strategy in terms of embedding capacity and image quality. In the first experiment, we embedded the simulation data produced by a pseudo-random bit generator into the cover images. In contrast, in the second experiment, some real data, which were secret images with resolutions of 256×256 pixels as shown in Fig. 8, were embedded into the Lena cover image. Besides, another experiment was designed to test the anti-detection performance of the multi-group method. In this experiment, 1000 512×384 uncompressed images from the UCID uncompressed image database [26] were used as the test images, and they were converted into grayscale ones before embedding.

We used the peak signal to noise ratio ($PSNR$) to evaluate image quality between the cover image and the stego image, $PSNR$ is defined as shown, and the units are dB:

$$PSNR=10\times\log_{10}\left(\frac{255^2}{MSE}\right). \tag{11}$$

MSE is the mean squared error between the cover image CI and the stego image SI in the size of $H\times W$, where H and W represent the height and width of the image:

$$MSE=\frac{1}{H\times W}\times\sum_{i=1}^{H}\sum_{j=1}^{W}(CI(i,j)-SI(i,j))^2. \tag{12}$$

4.1. The experiment for simulation secret data embedding

In this experiment, first, we compared the embedding performance of our method, EMD method, and the

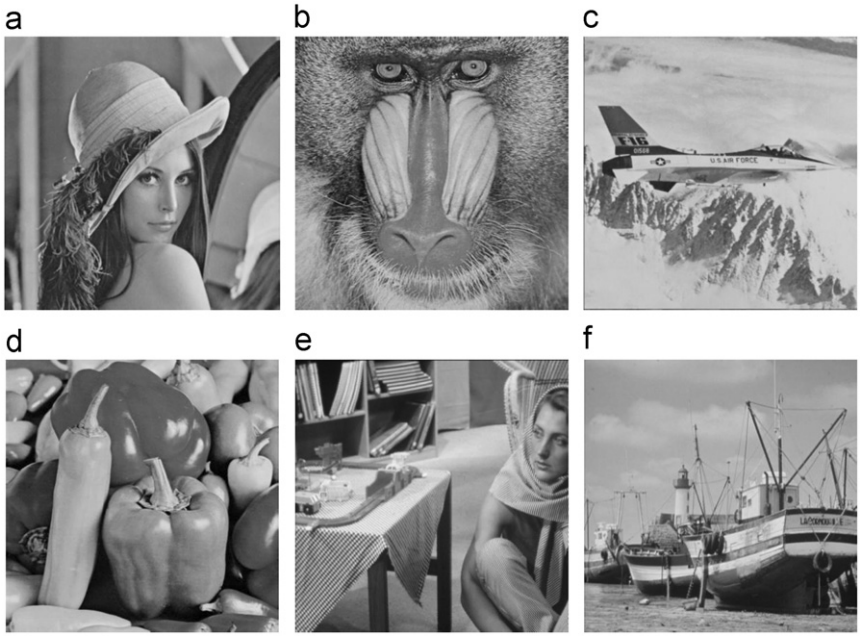

Fig. 7. Six grayscale cover images with sizes of 512×512 pixels. (a) Lena; (b) Baboon; (c) Airplane; (d) Pepper; (e) Barbara and (f) Boat.

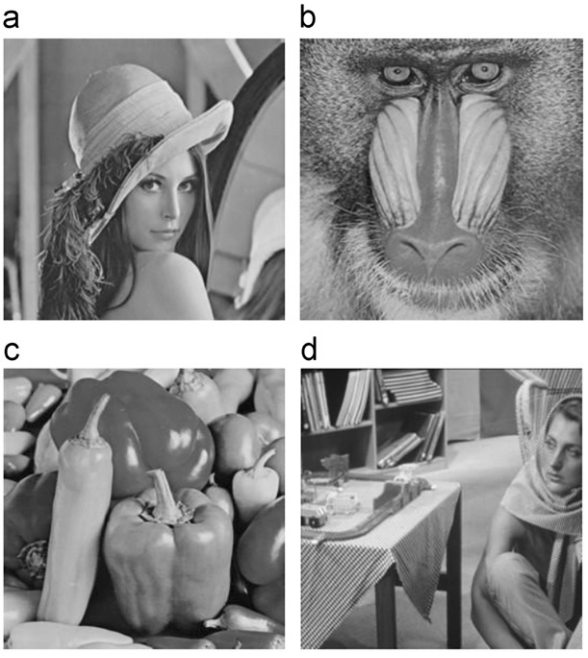

Fig. 8. Four grayscale secret images with sizes of 256×256 pixels. (a) Lena; (b) Baboon; (c) Pepper and (d) Barbara.

section-wise strategy over six test cover images with the maximal embedding capacity as shown in Table 1. For the EMD method, we set the parameters $K=2$ and $n=2$. In the section-wise strategy, we set the range of SP as [0,4] and the range of DP as [0,6] as suggested in the researchers' paper. In our multi-group method, two cases, $k=1$ and $k=2$, were discussed to make a thorough analysis and a reasonable comparison.

In Table 1, it can be seen that the embedding capacity of the multi-group method is greater than that of the EMD method, k was 1 or 2, and the image quality of the multi-group method is not lower that of the EMD method. Particularly, in the case of $k=2$, $n_1=n_2=2$, the average embedding capacity of the multi-group method is 40,000 bits more than that of the EMD method. It is worthwhile to note that our method is superior to EMD method with respect to embedding capacity as well as image quality when $k=2$, $n_1=2$ and $n_2=3$. The reason is that, when $K=2$ and $n=2$ in EMD, four secret bits are embedded into two pixel-groups, so two of the four pixels may be modified in the worst case scenario. However, four or five secret bits are embedded into two pixel-groups but, at most, two of the five pixels may be modified in the multi-group if $n_1=2$ and $n_2=3$. For the section-wise strategy, with the similar *PSNR*, the average embedding capacity is 4600 bits less than the multi-group method when $k=2$, $n_1=2$ and $n_2=3$. In the other cases, since the embedding capacity of the section-wise strategy is similar to that of the EMD method, the capacity of our method is also greater than that of the section-wise strategy.

The superiority of the multi-group method can be verified by mathematically. The practical embedding rate of the multi-group method can be calculated by Eq. (13):

$$R_{multi-group} = \frac{P(S|L) + \left\lfloor \log_2 \prod_{i=1}^{k} (2n_i+1) \right\rfloor}{\sum_{i=1}^{k} n_i}. \tag{13}$$

$P(S|L)$ stands for the probability of the extracted L-bits secret data S, which can be represented by the switch map, where $L = \left\lceil \log_2 \Pi_{i=1}^{k} (2n_i+1) \right\rceil$. Assume that the secret data embedded into all pixel-pieces can be classified as Q L-bit secret streams and W $(L-1)$-bit secret

Table 1
Results of the comparison of capacity and *PSNR* for six cover images.

Images	EMD		Section-wise		Multi-group $k=1$, $n=2$		Multi-group $k=2$, $n_1=2$, $n_2=2$		Multi-group $k=2$, $n_1=2$, $n_2=3$	
	Capacity (bits)	*PSNR* (dB)	Capacity (bits)	*PSNR* (dB)	Capacity (bits)	*PSNR* (dB)	Capacity (bits)	*PSNR* (dB)	Capacity (bits)	*PSNR* (dB)
Airplane	262144	52.11	262140	52.92	288569	52.11	302555	52.11	266706	52.92
Baboon	262144	52.11	262140	52.92	288756	52.11	302773	52.11	266690	52.92
Barbara	262144	52.10	262140	52.91	288450	52.12	302591	52.11	266699	52.93
Boat	262144	52.11	262140	52.93	288659	52.11	302779	52.12	266695	52.94
Lena	262144	52.11	262140	52.92	288600	52.11	302714	52.11	266780	52.93
Pepper	262144	52.10	262140	52.92	288409	52.12	302786	52.11	266815	52.93

streams, so $P(S|L)$ can be determined by Eq. (14):

$$P(S|L)=\frac{Q}{Q+W}. \tag{14}$$

In two special cases, i.e., $Q=0$ and $W=0$, the theoretical minimum value and theoretical maximum value can be represented by Eqs. (15) and (16), respectively:

$$(R_{multi-group})_{\min}=\frac{\left\lfloor \log_2 \Pi_{i=1}^{k}(2n_i+1)\right\rfloor}{\sum_{i=1}^{k} n_i}. \tag{15}$$

$$(R_{multi-group})_{\max}=\frac{1+\left\lfloor \log_2 \Pi_{i=1}^{k}(2n_i+1)\right\rfloor}{\sum_{i=1}^{k} n_i}. \tag{16}$$

In the multi-group method, if $n_1=n_2=\ldots=n_i=n$, Eq. (13) can be transformed into $R_{multi-group}=((P(S|L)+\lfloor k\times\log_2(2n+1)\rfloor)/(k\times n))$. According to the embedding rate of EMD as determined by Eq. (7), we can find that the embedding rate of the multi-group method will never be less than that of EMD if $k=K$ in the condition of each pixel-group containing the same pixel number, i.e. $R_{multi-group}\geq R_{EMD}$. If $R_{multi-group}=R_{EMD}$, it means that $P(S|L)=0$ and that the secret data embedded into all pixel-pieces are entirely $(L-1)$-bit secret streams, i.e. $Q=0$. Otherwise, our method is superior to EMD. Further, the practical embedding rate of the multi-group method must be greater than that of EMD, and we can stop considering whether $k=K$ if the following condition is satisfied:

$$P(S|L)\geq k\times\log_2(2n+1)-\lfloor k\times\log_2(2n+1)\rfloor. \tag{17}$$

The embedding rate of the section-wise strategy can be represented as:

$$R_{section-wise}=\frac{\lfloor \log_2((2n_1+1)\times(2n_2+1))\rfloor}{n_1+n_2}. \tag{18}$$

If $k=2$, Eq. (13) can be transformed into

$$R_{multi-group}=\frac{P(S|L)+\lfloor \log_2((2n_1+1)\times(2n_2+1))\rfloor}{n_1+n_2}.$$

Similar to the relationship between multi-group and EMD, we can prove $R_{multi-group}\geq R_{section-wise}$. The mathematical justification verifies the truth explained in the Section 3.3.

Next, the comparisons of embedding performance between our multi-group method and the EMD method and section-wise strategy, with an increased number of pixels, are shown in Fig. 9 and Table 2. In the EMD method, we set the pixel number of each pixel-group from 2 to 10. In the section-wise strategy, the combination of the numbers of pixels in the selective group and in the descriptive group was set to $(n, n+1)$. In the multi-group method, we also used the case of $k=1$ and $k=2$. According to the above explanation, a similar scenario can be observed in Table 2 for the Lena image with increasing n. Fig. 9(a)–(f) visually and clearly display the superiority of the multi-group method, confirming the mathematical proof. In addition, Fig. 9(a)–(f) show that the results are almost identical, demonstrating that the complexity and content of an image does not influence the embedding performance of the multi-group method significantly.

4.2. The experiment for real secret data embedding

In this experiment, we used real images with sizes of 256×256 pixels as secret data to research the embedding performance of the multi-group method. We knew that our method was not sensitive to the complexity and content of an image, so we just selected the Lena image as the cover image. Fig. 10 displays the results of the comparison. In some cases, the embedding capacity may increase or decrease marginally when experimental results are compared with the simulated embedding of secret data. These differences are due to the coherence of real data, and the multi-group method is still able to hide 33,000 more real secret bits than the EMD method and the section-wise strategy in the maximal embedding capacity, and it also provided better image quality than the EMD method. In summation, our method has the evident advantages over the other two methods.

4.3. Anti-detection test

Anti-detection test is a very important indicator for evaluating the embedding performance of steganography. In this experiment, we used the calibrated adjacent HCF-COM detector [27], which was proposed by Andrew D. Ker, to compare the anti-detection performance on the multi-group method, the EMD method, and the section-wise strategy. The image Lena with size of 256×256 pixels was used as the secret data.

We give the receiver operating characteristic (ROC) curves, showing how the tradeoff of detection and false positive as the detection threshold is varied, for the 1000

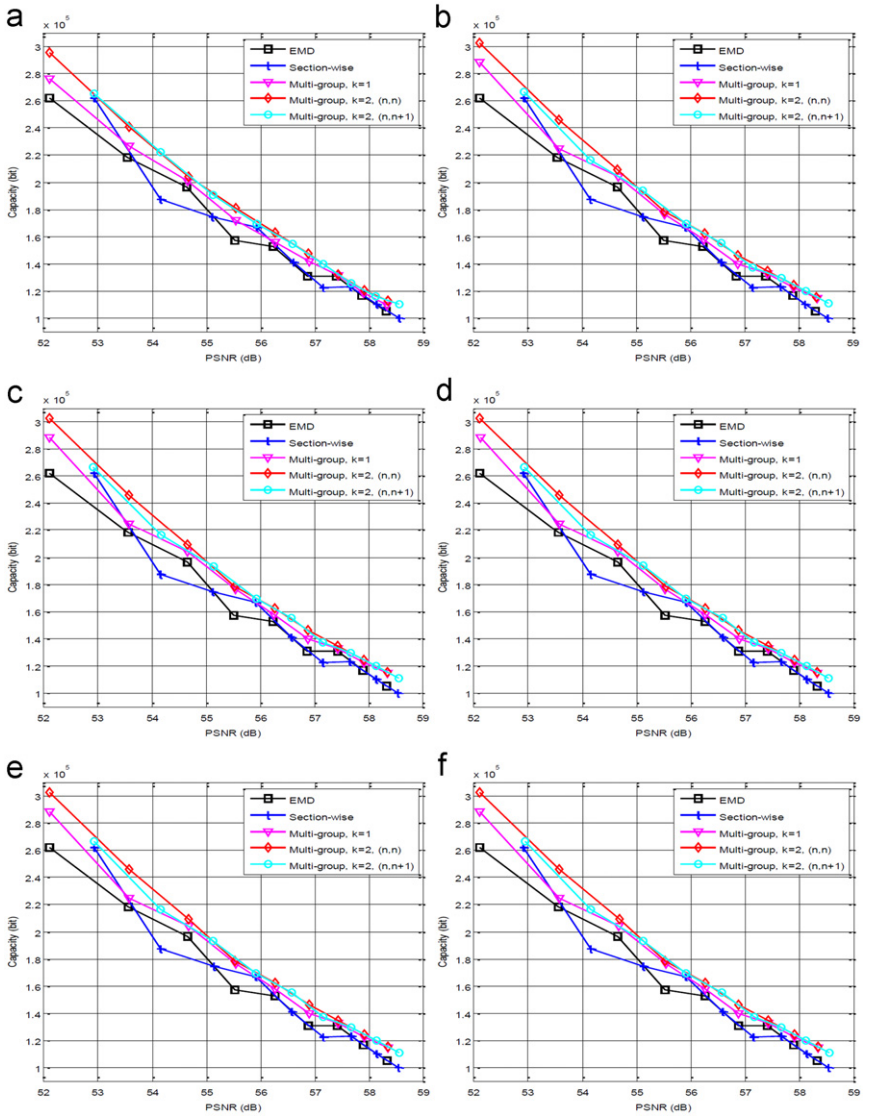

Fig. 9. Embedding performance with increase of pixel number over six cover images. (a) Lena; (b) Baboon; (c) Airplane; (d) Pepper; (e) Barbara and (f) Boat.

Table 2
Results of the comparison of capacity and *PSNR* for the Lena image.

n	EMD		Section-wise		Multi-group $k=1$		Multi-group $k=2$, $n_1=n_2=n$		Multi-group $k=2$, $n_1=n, n_2=n+1$	
	Capacity (bits)	*PSNR* (dB)	Capacity (bits)	*PSNR* (dB)	Capacity (bits)	*PSNR* (dB)	Capacity (bits)	*PSNR* (dB)	Capacity (bits)	*PSNR* (dB)
2	262144	52.10	262140	52.92	288600	52.11	302714	52.11	266780	52.93
3	218450	53.54	187245	54.15	224903	53.58	245957	53.58	216665	54.16
4	196608	54.64	174762	55.12	203847	54.66	209356	54.66	193450	55.13
5	157284	55.52	166817	55.91	176350	55.53	178542	55.54	169382	55.91
6	152915	56.22	141148	56.59	158092	56.26	162094	56.27	155364	56.58
7	131068	56.85	122332	57.16	139980	56.88	146319	56.88	137529	57.16
8	131072	57.41	123360	57.67	133056	57.43	134646	57.43	129540	57.69
9	116508	57.88	110376	58.13	122761	57.91	124640	57.91	120079	58.13
10	104856	58.31	99864	58.54	114963	58.35	115136	58.34	110848	58.55

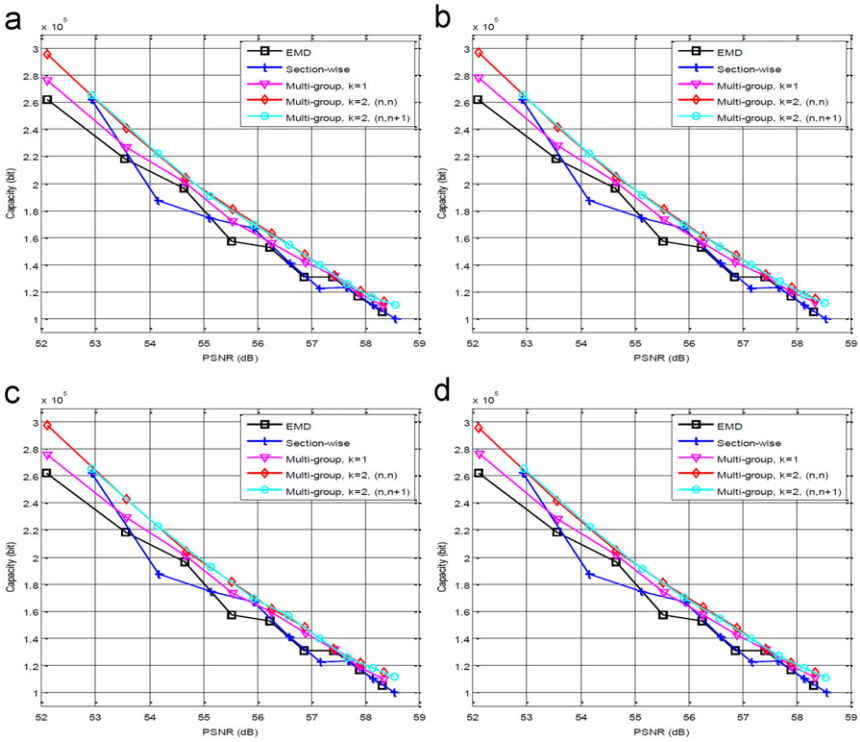

Fig. 10. Embedding performance with increase of pixel number for four embedded secret images. (a) Lena; (b) Baboon; (c) Pepper and (d) Barbara.

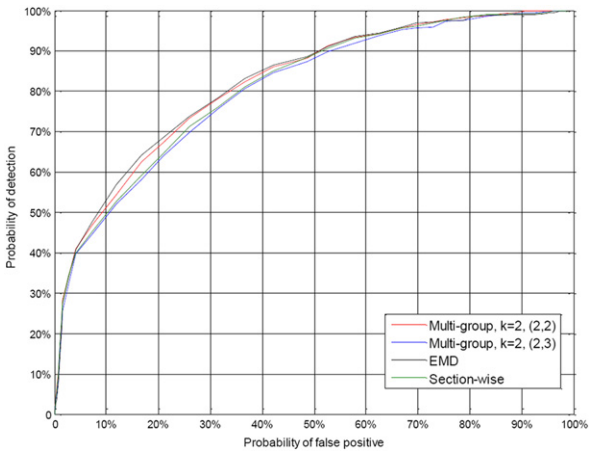

Fig. 11. ROC curves with a payload of 1.00 bpp.

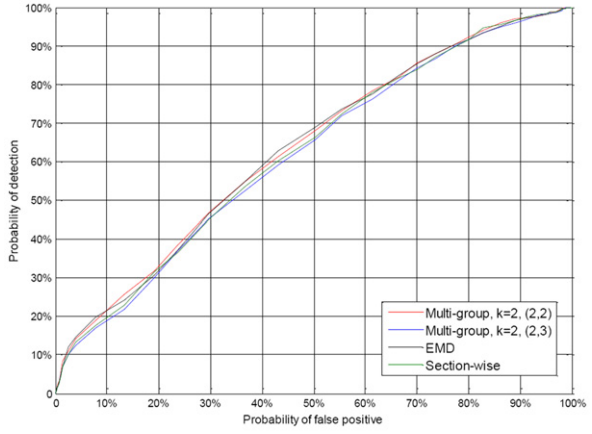

Fig. 12. ROC curves with a payload of 0.50 bpp.

UCID images with the payload of 1.00 bpp in Fig. 11. In Fig. 12, the payloads are 0.50 bpp.

At the same probability of false positive, the multi-group method with $k=2$, $n_1=2$ and $n_2=3$ has lower probability of detection than other methods as shown in Figs. 11 and 12. Generally, the probability of detection of the multi-group method with $k=2$, and $n_1=n_2=2$ is lower than that of the EMD method. It means that fewer modifications are introduced by the multi-group method and the stego image produced by the proposed method is more difficult to be detected.

The greatest ascendancy of the multi-group method is that it can provide larger payload while maintaining the image quality similar to those of the EMD method and the section-wise strategy. Therefore, in this experiment, the stego images produced by the proposed method have many unused pixels. Take 1.00 bpp for instance, average 2331 pixels are left after embedding by the multi-group method with $n_1=2$ and $n_2=3$, and the average number of vacant pixels is 21364 using $n_1=2=n_2=2$. Consequently, the ROC of the proposed method may be improved further if the embedding algorithm was designed elaborately for

scattering the payload of 1.00 bpp into the cover image. The purpose of that is to heighten the close intensities of adjacent pixels, then to increase the difficulty of detection of the calibrated adjacent HCF-COM detector.

5. Conclusions

In this paper, we proposed an adjustable multi-group method based on switch map to improve the embedding efficiency of the EMD method and the section-wise strategy. In our method, the switch map, which is constructed before the data embedding procedure takes place, can exploit all modification directions to eliminate the two styles of redundancy problems of the EMD method and the section-wise strategy, and increase the probability of achieving larger embedding capacity. According to the switch map that was constructed, L-bit secret data or $(L-1)$-bit secret data can be embedded into the combinations of some pixel-groups with less pixel changes, thereby enhancing the image quality.

As the experimental results revealed, with simple calculations, our proposed method can achieve larger embedding capacity while providing better image quality when compared to the EMD method and the section-wise strategy. These conclusions are supported by results achieved from the experimental embedding of simulation data and real data, and mathematical justification. In addition, we can adjust the pixel numbers of each pixel-group freely to embed data for different actual applications.

References

[1] R. Davis, The data encryption standard in perspective, IEEE Communications Magazine 16 (6) (1978) 5–9.
[2] R.L. Rivest, A. Shamir, L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, Communications of the ACM 21 (2) (1978) 120–126.
[3] H.J. Highland, Data encryption: a non-mathematical approach, Computers & Security 16 (5) (1997) 369–386.
[4] W. Stallings, Cryptography and Network Security: Principles and Practice, third ed., Pearson Education, New Jersey, 2003.
[5] F.A.P. Petitcolas, R.J. Anderson, M.G. Kuhn, Information hiding-a survey, Proceedings of the IEEE 87 (7) (1999) 1062–1078.
[6] I.J. Cox, M.L. Miller, J.A. Bloom, J. Fridrich, T. Kalker, Digital Watermarking and Steganography, second ed., Morgan Kaufmann, 2008.
[7] S. Katzenbeisser, F.A.P. Petitcolas, Information Hiding Techniques for Steganography and Digital Watermarking, Artech House, Boston, 2000.
[8] C.K. Chan, L.M. Cheng, Hiding data in images by simple LSB substitution, Pattern Recognition 37 (3) (2004) 469–474.
[9] R.Z. Wang, C.F. Lin, J.C. Lin, Image hiding by optimal LSB substitution and genetic algorithm, Pattern Recognition 34 (3) (2001) 671–683.
[10] S. Dumitrescu, X. Wu, Z. Wang, Detection of LSB steganography via sample pair analysis, IEEE Transactions on Signal Processing 51 (7) (2003) 1995–2007.
[11] A.D. Ker, Improved detection of LSB steganography in grayscale images, in: J. Fridrich (Ed.), Proceedings of the Sixth International Workshop on Information Hiding, vol. 3200, Canada, 2004, pp. 97–115.
[12] J. Fridrich, M. Goljan, D. Rui, Detecting steganography in color and grayscale images, IEEE Multimedia 8 (4) (2001) 22–28.
[13] J. Mielikainen, LSB matching revisited, IEEE Signal Processing Letters 13 (5) (2006) 285–287.
[14] X. Zhang, S. Wang, Efficient steganographic embedding by exploiting modification direction, IEEE Communications Letters 10 (11) (2006) 781–783.
[15] X. Zhang, W. Zhang, S. Wang, Efficient double-layered steganographic embedding, Electronics Letters 43 (8) (2007) 482–483.
[16] W. Zhang, S. Wang, X. Zhang, Improving embedding efficiency of covering codes for applications in steganography, IEEE Communications Letters 11 (8) (2007) 680–682.
[17] C.C. Chang, W.L. Tai, K.N. Chen, Improvements of EMD embedding for large payloads, Proceedings of the Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIHMSP07), Kaohsiung, Taiwan, 2007, pp. 473–476.
[18] C.F. Lee, Y.R. Wang, C.C. Chang, A. Steganographic Method with high embedding capacity by improving exploiting modification direction, Proceedings of the Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIHMSP07), Kaohsiung, Taiwan, 2007, pp. 497–500.
[19] W. Zhang, X. Zhang, S. Wang, A Double Layered "Plus–Minus One" data embedding scheme, IEEE Signal Processing Letters 14 (11) (2007) 848–851.
[20] R.M. Chao, H.C. Wu, C.C. Lee, Y.P. Chu, A novel image data hiding scheme with diamond encoding, EURASIP Journal on Information Security doi:10.1155/2009/658047 (2009).
[21] K.H. Jung, K.Y. Yoo, Improved exploiting modification direction method by modulus operation, International Journal of Signal Processing, Image Processing and Pattern 2 (1) (2009) 79–88.
[22] C.C. Chang, C.F. Lee, L.Y. Chuang, Using dynamic programming strategy to find an optimal solution to exploiting modification direction embedding method, Proceedings of the Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIHMSP07), Kaohsiung, Taiwan, 2007, pp. 489–492.
[23] C.F. Lee, C.C. Chang, K.H. Wang, An improvement of EMD embedding method for large payloads by pixel segmentation strategy, Image and Vision Computing 26 (12) (2008) 1670–1676.
[24] T.D. Kieu, C.C. Chang, A steganographic scheme by fully exploiting modification directions, Expert Systems with Applications 38 (8) (2011) 10648–10657.
[25] J. Wang, Y. Sun, H. Xu, K. Chen, H.J. Kim, S.H. Joo, An improved section-wise exploiting modification direction method, Signal Processing 90 (11) (2010) 2954–2964.
[26] G. Schaefer, M. Stich (2004), UCID: an uncompressed colour image database, Proceedings of SPIE, Storage and Retrieval Methods and Applications for Multimedia 2004, San Jose, USA, pp. 472–480.
[27] A.D. Ker, Steganalysis of LSB matching in grayscale images, IEEE Signal Processing Letters 12 (6) (2005) 441–444.