Expert Systems with Applications 38 (2011) 10648–10657

Contents lists available at ScienceDirect

Expert Systems with Applications

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

A steganographic scheme by fully exploiting modification directions

The Duc Kieu [a,*], Chin-Chen Chang [b]

[a] School of Computer Science and Engineering, International University, Vietnam National University, Ho Chi Minh City, Viet Nam
[b] Department of Information Engineering and Computer Science, Feng Chia University, Taichung 40724, Taiwan, ROC

ARTICLE INFO

Keywords:
Information hiding
Watermarking
Steganography
Secret communication
Modification directions

ABSTRACT

Recently, Zhang and Wang proposed a steganographic scheme by exploiting modification direction (EMD) to embed one secret digit d in the base-$(2 \times n + 1)$ notational system into a group of n cover pixels at a time. Therefore, the hiding capacity of the EMD method is $\log_2(2 \times n + 1)/n$ bit per pixel (bpp). In addition, its visual quality is not optimal. To overcome the drawbacks of the EMD method, we propose a novel steganographic scheme by exploiting eight modification directions to hide several secret bits into a cover pixel pair at a time. By this way, the proposed method can achieve various hiding capacities of 1, 2, 3, 4, and 4.5 bpp and good visual qualities of 52.39, 46.75, 40.83, 34.83, and 31.70 dB, respectively. The experimental results show that the proposed method outperforms three recently published works, namely Mielikainen's, Zhang and Wang's, and Yang et al.'s methods.

1. Introduction

The proliferation of network technologies and digital devices makes digital multimedia delivery fast and easy. However, distributing digital data over public networks such as the Internet is not really safe due to copy violation, counterfeiting, forgery, and fraud. Therefore, protective methods for digital data, especially for sensitive data, are highly essential. Conventionally, secret data can be protected by cryptographic methods such as DES (Davis, 1978) or RSA (Rivest, Shamir, & Adleman, 1978). The drawback of cryptography is that cryptography can secure secret data in transit, but once they have been decrypted, the content of the secret data has no further protection (Cox, Bloom, Kalker, 2007). Alternatively, confidential data can be protected by using information hiding techniques. An information hiding system hides secret information into a cover object (e.g., an image, audio, video, or written text) to obtain an embedded object (also called a watermarked object in watermarking applications or a stego object in steganographic applications). For more secure, a cryptographic technique can be applied to an information hiding scheme to encrypt the secret data prior to embedding.

In general, information hiding (also called data hiding or data embedding) includes digital watermarking and steganography (Petitcolas, Anderson, & Kuhn, 1999). Watermarking is used for copyright protection, broadcast monitoring, and transaction tracking. A watermarking scheme imperceptibly alters a cover object to embed a message about the cover object (e.g., owner's identifier) (Cox et al., 2007). The robustness (i.e., the ability to resist certain malicious attacks such as common signal processing operations) of digital watermarking schemes is critical. In contrast, steganography is used for secret communications. A steganographic method undetectably alters a cover object to conceal a secret message (Cox et al., 2007). Thus, steganographic methods can hide the very presence of covert communications.

Data hiding techniques can be carried out in three domains (Langelaar, Setyawan, & Lagendijk, 2000), namely, spatial domain (Mielikainen, 2006), compressed domain (Chang, Kieu, & Chou, 2009a; Chang, Kieu, & Wu, 2009; Chang, Tai, & Lin, 2006), and frequency domain (Lee, Yoo, & Kalker, 2007). Each domain has its own advantages and disadvantages in regard to hiding capacity, execution time, and storage space. The fundamental requirements of information hiding systems are good visual quality (i.e., image quality), high hiding capacity, robustness, and steganographic security (i.e., statistically undetectable) (Langelaar et al., 2000).

Designing a new data hiding system achieving good visual quality, high hiding capacity, robustness, and steganographic security is a technically challenging problem. Thus, there are different approaches in designing data hiding systems in the literature. Some of these approaches are as follows. The first approach is to increase hiding capacity (also called embedding capacity or payload) while maintaining a good visual quality or at the cost of lower visual quality (Lan & Tewfik, 2006). This approach is appropriate to applications where high hiding capacity is desired. The second approach purposes to devise a robust data hiding scheme (Ni et al., 2008). This design serves robust watermarking systems. The third

* Corresponding author. Address: School of Computer Science and Engineering, International University, Vietnam National University, Block 6, Linh Trung Ward, Thu Duc District, Ho Chi Minh City, Viet Nam. Tel.: +84 8 37244270x3232; fax: +84 8 37244271.
E-mail addresses: ktduc@hcmiu.edu.vn, ktduc0323@yahoo.com.au (T.D. Kieu), ccc@cs.ccu.edu.tw (C.-C. Chang).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.02.122

approach aims at enhancing visual quality while keeping the same hiding capacity or at the cost of lower hiding capacity (Ni, Shi, Ansari, & Su, 2006). The fourth approach intends to devise a data hiding scheme with high embedding efficiency (Fridrich, Goljian, & Soukal, 2006; Mielikainen, 2006; Westfeld, 2001). This approach can increase the steganographic security of a data hiding scheme because it is less detectable by statistical steganalysis (Fridrich et al., 2007).

A simple data hiding system in the spatial domain is the least significant bit (LSB) replacement method proposed by Turner (1989). The LSB replacement method (also called the LSB substitution) directly embeds k secret bits into k least significant bits (LSBs) of each cover pixel, where $1 \leqslant k \leqslant 8$. In general, the LSB replacement (LSB-R) method can achieve an acceptable visual quality when $k \leqslant 3$. For $4 \leqslant k \leqslant 8$, the visual quality the LSB-R scheme is severely degraded. The LSB replacement method is inherently asymmetric. That is, an even-valued pixel will either retain its value or be incremented by one. However, it will never be decremented. The converse is true for an odd-valued pixel. This means that an odd-valued pixel will either remain unchanged or be decremented by one. Nevertheless, it will never be incremented. This asymmetry is exploited for the steganalytic purpose (i.e., steganalysis). It is known that the LSB-R method is easily detected by some detectors (Fridrich, Goljan, & Du, 2001; Harmsen and Pearlman, 2003). Then, many data hiding schemes were proposed to improve the LSB-R method in terms of visual quality and/or hiding capacity by using genetic algorithm (Wang, Lin, & Lin, 2001), dynamic programming (Chang, Hsiao, & Chan, 2003), pixel-value differencing (PVD) Wu & Tsai, 2003; Wu et al., 2005, and optimal pixel adjustment process (OPAP) Chan & Cheng, 2004.

To overcome the asymmetry of the LSB replacement method, Sharp proposed a data hiding scheme called the LSB matching method (Sharp, 2001). The LSB matching (LSB-M) method does not simply replace an LSB of a cover pixel with a secret bit. Instead, if the secret bit does not match the LSB of the cover pixel, then the cover pixel is randomly either incremented or decremented by one. Therefore, the asymmetry of odd- and even-valued pixels is eliminated. As a result, the detection of the LSB-M method by statistical detectors is known to be much more difficult than detecting the LSB-R method (Ker, 2004). However, the LSB-M method is then detected by the detector proposed by Ker (2005). It is noted that the LSB-M method aims to remove the asymmetry of the LSB-R method so the LSB-M method has the same visual quality and hiding capacity as the LSB-R method.

To further enhance the LSB-M method in terms of visual quality, Mielikainen offered a data embedding scheme called the LSB matching revisited (Mielikainen, 2006). In the LSB matching revisited (LSB-M-R) scheme, the binary function and four embedding rules are used to embed two secret bits into a cover pixel pair at a time. The main purpose of Mielikainen's method is to embed the same payload as the LSB-M method (i.e., the hiding capacity of 1 bit per pixel (bpp)) but fewer changes to the cover image. Specifically, the expected number of modifications per pixel (ENMPP) of Mielikainen's scheme is 0.375 whereas that of the LSB-M method is 0.5. Consequently, the visual quality measured by peak signal-to-noise ratio (PSNR) of the LSB-M-R method is better than that of the LSB-M method. The details of embedding and extracting processes of Mielikainen's scheme can be found in Mielikainen (2006).

Zhang and Wang claimed that the modification directions of Mielikainen's scheme are not explored fully. To fully exploit the modification directions of Mielikainen's scheme, they proposed the steganographic scheme by exploiting modification direction (EMD) (Zhang & Wang, 2006). The EMD method embeds one secret digit in the base-$(2 \times n + 1)$ notational system into a group of n cover pixels at a time, where n is an integer greater than 1. Theoretically, the hiding capacity of the EMD method is $\log_2(2 \times n + 1)/n$ bpp. Practically, it is clear that this method achieves its maximum hiding capacity of 1 bpp when n equals 2. Actually, the visual quality of the EMD method at its maximum hiding capacity of 1 bpp (i.e., $n = 2$) is not optimal. Specifically, the PSNR value of the EMD method is slightly smaller than that of Mielikainen's scheme at hiding capacity of 1 bpp. This is because the ENMPP of the EMD method is 0.4. In addition, for $n = 2$, Zhang and Wang's method only utilizes four modification directions, namely East (**E**, →), North (**N**, ↑), West (**W**, ←), and South (**S**, ↓) to conceal each secret digit into a group of two consecutive cover pixels. In late 2008, Yang et al. proposed the adaptive LSB replacement (A-LSB-R) steganographic scheme (Yang, Weng, Wang, & Sun, 2008) to provide a larger embedding capacity (i.e., around 4 bpp) and higher image quality compared to Wu et al.'s method (2005). To improve Zhang and Wang's method in the case of $n = 2$ in terms of the ENMPP value (and so visual quality) and provide a data hiding scheme with various hiding capacities from 1 bpp up to 4.5 bpp (i.e., $\lfloor log_2 s^2 \rfloor/2$ bpp, where s is an integer greater than or equal to 2 and the notation $\lfloor x \rfloor$ is the floor function meaning the greatest integer less than or equal to x), we propose a novel steganographic scheme by utilizing eight modification directions, namely East (**E**, →), North-East (NE, ↗), North (N, ↑), North-West (NW, ↖), West (**W**, ←), South-West (NW, ↙), South (S, ↓), and South-East (NW, ↘) for embedding several secret bits into a block of two cover pixels at a time. To achieve the goal of designing our new steganographic scheme, we propose a novel extraction function (also called the modified extraction function) by modifying the extraction function proposed by Zhang and Wang (2006). The modified extraction function allows the proposed method to exploit eight modification directions for embedding secret data, restrict the embedding distortion into a square of various sizes (e.g., 2×2, 3×3, and so on), and use the minimum distortion embedding (MDE) process. By this way, the proposed method can achieve various hiding capacities and good visual qualities compared to three recently published works, namely Mielikainen's method (2006), Zhang and Wang's method (2006), and Yang et al.'s method (2008).

The remaining of this paper is organized as follows. The review of Zhang and Wang's scheme is presented in Section 2. Our proposed method is detailed in Section 3. The experimental results and discussion are shown in Section 4. Finally, some conclusions are made in Section 5.

2. Zhang and Wang's method

Zhang and Wang proposed the steganographic scheme by exploiting modification direction (EMD) Zhang & Wang, 2006. Let us denote the grayscale cover image X sized $H \times W$ as $X = \{x_i \mid 1 \leqslant i \leqslant H \times W, \quad x_i \in [0, 255]\}$, the binary secret message $M = (m_1 m_2 \ldots m_{LM})$ is of length LM, and the grayscale stego image Y sized $H \times W$ as $Y = \{y_i \mid 1 \leqslant i \leqslant H \times W, y_i \in [0, 255]\}$. In general, the EMD method embeds one secret digit d in the base-$(2 \times n + 1)$ notational system into a group of n cover pixels in the cover image X at a time. Theoretically, the hiding capacity of the EMD method is $\log_2(2 \times n + 1)/n$ bpp (i.e., about 1.161 bpp). Practically, it is clear that this method achieves its maximum hiding capacity of 1 bpp when n equals 2.

The EMD method is now intuitively presented for the case of $n = 2$. This is the case that the EMD method achieves its maximum hiding capacity. Firstly, the extraction function f, which is defined by Eq. (1), is used to generate the matrix R sized 256×256.

$$f(x_i, x_{i+1}) = x_i + 2 \times x_{i+1} \bmod 5, \tag{1}$$

where x_i and x_{i+1} are grayscale values (i.e., $0 \leqslant x_i, x_{i+1} \leqslant 255$). A portion of the matrix R is shown in Fig. 1. The element located at the x_ith row and x_{i+1}th column of the matrix R is denoted by $R[x_i][x_{i+1}]$. It can be seen from Fig. 1 that the matrix R has an interesting

x_i \ x_{i+1}	0	…	11	12	13	14	15	16	17	18	…	255
0	0	…	2	4	1	3	0	2	4	1	…	
:	:	:	:	:	:	:	:	:	:	:		
11	1	…	3	0	2	4	1	3	0	2	…	
12	2	…	4	1	3	0	2	4	1	3	…	
13	3	…	0	2	4	1	3	0	2	4	…	
14	4	…	1	3	0	2	4	1	3	0	…	
15	0	…	2	4	1	3	0	2	4	1	…	
16	1	…	3	0	2	4	1	3	0	2	…	
17	2	…	4	1	3	0	2	4	1	3	…	
18	3	…	0	2	4	1	3	0	2	4	…	
:	:	…	:	:	:	:	:	:	:	:	…	
255												

Fig. 1. The matrix *R*.

property, namely any five neighboring elements $R[x_i][x_{i+1}]$, $R[x_i][x_{i+1}+1]$, $R[x_i-1][x_{i+1}]$, $R[x_i][x_{i+1}-1]$, and $R[x_i+1][x_{i+1}]$ along the horizontal and vertical directions are different digits in the base-5 numeral system. This property is exploited by the EMD method to embed the secret message *M*. Secondly, the binary secret message *M* is partitioned into the segments of four bits. Next, each 4-bit segment is converted into two secret digits in the 5-ary numeral system.

Thirdly, the grayscale cover image *X* is divided into non-overlapping groups of two consecutive cover pixels. Then, each secret digit *d* in the base-5 numeral system (also called a secret digit for short) is embedded into one cover pixel pair (x_i, x_{i+1}) at a time by increasing or decreasing only one cover pixel in the pair by 1, where $i \in \{1, 3, \ldots, H \times W - 1\}$. More specifically, if $d = R[x_i][x_{i+1}]$, then the stego pixel pair is computed by $(y_i, y_{i+1}) = (x_i, x_{i+1})$, where $(a, b) = (c, d)$ means that $a = c$ and $b = d$. Otherwise, search from the 3×3 sized square *B* centered at $R[x_i][x_{i+1}]$ along four directions (i.e., East, North, West, and South) to find out the element equal to *d*. The 3×3 square *B* contains four candidate elements, namely $R[x_i][x_{i+1}+1]$, $R[x_i-1][x_{i+1}]$, $R[x_i][x_{i+1}-1]$, and $R[x_i+1][x_{i+1}]$. Let us denote the found element as $R[u][v]$, where $u \in \{x_i - 1, x_i + 1\}$ and $v \in \{x_{i+1} - 1, x_{i+1} + 1\}$. Then, the stego pixel pair is computed by $(y_i, y_{i+1}) = (u, v)$. It is noted that only either x_i or x_{i+1} needs to be modified at most by 1 or x_i and x_{i+1} are left unchanged to obtain the stego pixel pair (y_i, y_{i+1}). This embedding process is repeated until all secret digits are embedded into the cover image *X* to obtain the stego image *Y*.

At the receiving side, with the received stego image *Y*, an intended receiver can extract each embedded secret digit *d* from each stego pixel pair (y_i, y_{i+1}) in the stego image *Y* by $d = f(y_i, y_{i+1})$. The extracted secret digits *d*'s are gathered and converted back to the binary form to obtain the original secret message *M*.

3. The proposed scheme

Mielikainen's and Zhang and Wang's schemes (2006) are of the ±1 embedding scheme (Fridrich et al., 2007). That is, these schemes hide two secret bits into a cover pixel pair by increasing or decreasing only one cover pixel of the pair by at most 1. As mentioned in Section 2, due to aiming at achieving high embedding efficiency, Zhang and Wang only uses five neighboring elements $R[x_i][x_{i+1}]$, $R[x_i][x_{i+1}+1]$, $R[x_i-1][x_{i+1}]$, $R[x_i][x_{i+1}-1]$, and $R[x_i+1][x_{i+1}]$ along the horizontal and vertical directions to embed a secret digit into the cover pixel pair (x_i, x_{i+1}). Consequently, the four remaining neighboring elements $R[x_i-1][x_{i+1}-1]$, $R[x_i+1][x_{i+1}+1]$, $R[x_i-1][x_i+1]$, and $R[x_i+1][x_{i+1}-1]$ along the main diagonal and minor diagonal directions are not used for the embedding process. With the purpose of offering a steganographic scheme with various hiding capacities (i.e., $\lfloor \log_2 s^2 \rfloor / 2$ bpp, where *s* is an integer greater than or equal to 2 and the notation $\lfloor x \rfloor$ is the floor function meaning the greatest integer less than or equal to *x*) and the minimum embedding distortion, we propose the ±*r* embedding scheme, where $r = \lfloor s/2 \rfloor$ is a positive integer. That is, the proposed scheme embeds *k* secret bits, where $k = \lfloor \log_2 s^2 \rfloor / 2$, into a block of two cover pixels (x_i, x_{i+1}) by incrementing or/and decrementing x_i or/and x_{i+1} at most by *r* or leaving x_i and x_{i+1} intact to obtain the stego pixel pair (y_i, y_{i+1}). Especially, when *r* equals 1, the hiding capacity of the proposed method is the same as that of Mielikainen's and Zhang and Wang's schemes and the ENMPP of the proposed method is identical to that of Mielikainen's scheme (i.e., 0.375), which is better than that of the EMD method (i.e., 0.400).

3.1. The embedding phase

The proposed method embeds *k* secret bits $(m_1 m_2 \ldots m_k)$ of a binary secret message *M* into a cover pixel pair (x_i, x_{i+1}) of the grayscale cover image *X* at a time, where $i \in \{1, 3, \ldots, H \times W - 1\}$, to obtain a stego pixel pair (y_i, y_{i+1}) of the grayscale stego image *Y*. Before embedding secrets, the pixels in the cover image *X* are grouped into non-overlapping blocks of two pixels by a user-defined pairing rule. For example, the selection (also called selection rule (Fridrich et al., 2007) of two pixels into a pixel pair (x_i, x_{i+1}) can be done by using a pseudo-random number generator (PRNG) with a secret seed. This can increase the steganographic security of the proposed scheme. Firstly, the modified extraction function *F* of the proposed method is defined as

$$F(x_i, x_{i+1}) = [(s-1) \times x_i + s \times x_{i+1}] \bmod s^2, \quad (2)$$

where *s* is an integer greater than or equal to 2 and $0 \leqslant x_i, x_{i+1} \leqslant 255$. It is clear that the proposed extraction function *F* generates a number belonging to the set $\{0, 1, \ldots, s^2 - 1\}$.

Secondly, the proposed extraction function *F* is used to generate the mapping matrix *S* sized 256×256 (also called the matrix *S* for short). That is, according to Eq. (2), the element located at the x_ith row and x_{i+1}th column of the matrix *S* is specified by $S[x_i][x_{i+1}] = F(x_i, x_{i+1})$. The values of a part of the mapping matrix *S* for $s \in \{2, 3, 4, 6\}$ are shown in Fig. 2.

It can be observed from Fig. 2 that any $s \times s$ sized square in the matrix *S* contains different numbers in the s^2-ary numeral system. This interesting property of the matrix *S* generated by the proposed extraction function *F* is exploited to design our steganographic scheme. Thirdly, the value of *k* is computed by $k = \lfloor \log_2 s^2 \rfloor$, where the notation $\lfloor x \rfloor$ denotes the floor function returning the greatest integer less than or equal to *x*. The reason of selecting $k = \lfloor \log_2 s^2 \rfloor$ is to make sure that any *k*-bit number is equal to one of the numbers in the $s \times s$ sized square as shown in Fig. 2.

Fourthly, the parameter *r*, which is called the searching radius and calculated by $r = \lfloor s/2 \rfloor$ is to define the searching area used for embedding *k* secret bits into a cover pixel pair (x_i, x_{i+1}). The searching area is restricted into a $(2 \times r + 1) \times (2 \times r + 1)$ sized searching square centered at the element $S(x_i, x_{i+1})$ in the matrix *S*. The searching square is defined as

$$W_{(2\times r+1)\times(2\times r+1)}(s, (x_i, x_{i+1}), r) = \{S(x_i - r + u, x_{i+1} - r + v) \mid 0 \leqslant u \leqslant 2 \times r, 0 \leqslant v \leqslant 2 \times r, u \neq v\}. \quad (3)$$

For example, if the value of *s* is 2, then $r = 1$, and according to ** Eq. (3), the searching area is the 3×3 sized searching square $\boldsymbol{W}_{3\times3}(2, (x_i, x_{i+1}), 1) = \{S(x_i - 1, x_{i+1} - 1), S(x_i - 1, x_{i+1}), S(x_i - 1, x_{i+1} + 1), S(x_i, x_{i+1} - 1), S(x_i, x_{i+1} + 1), S(x_i + 1, x_{i+1} - 1), S(x_i + 1, x_{i+1}), S(x_i + 1, x_{i+1} + 1)\}$. As another demonstrative example, when $s = 4$, the searching square is $\boldsymbol{W}_{5\times5}(4, (4, 3), 2)$ shown in Fig. 3.

Next, read *k* secret bits $(m_1 m_2 \ldots m_k)$ from the binary secret message *M* and convert $(m_1 m_2 \ldots m_k)$ into a *k*-bit secret number *d* in the base-10 numeral system (also called the secret number for short). Then, the decimal number *d* is embedded into each pixel pair (x_i, x_{i+1}) of the cover image *X* at a time, where $i \in \{1, 3, \ldots, H \times W - 1\}$ as follows. If $d = S[x_i][x_{i+1}]$, then the stego pixel pair is calculated by

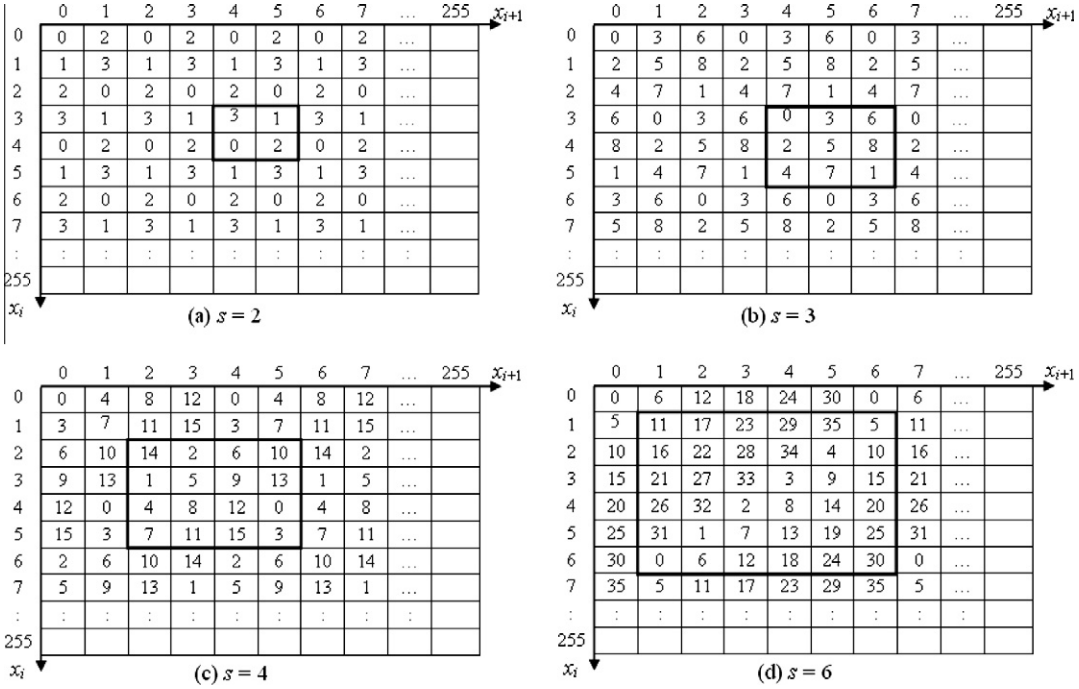

Fig. 2. Some values of the mapping matrix S for various values of s.

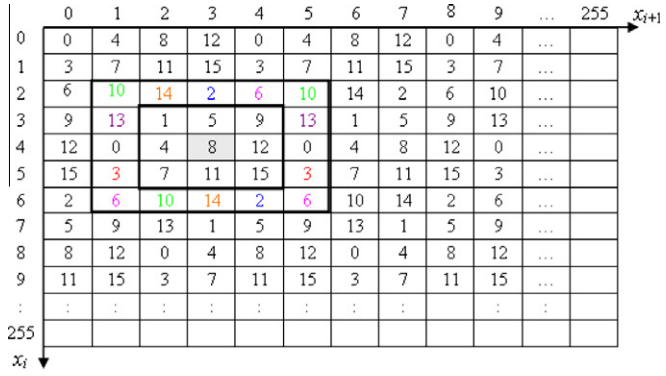

Fig. 3. The illustrative example of $W_{5\times5}(4,(4,3),2)$ for $s = 4$.

$(y_i, y_{i+1}) = (x_i, x_{i+1})$, where $(a, b) = (c, d)$ means that $a = c$ and $b = d$. Otherwise, search from the searching square $\boldsymbol{W}_{(2\times r+1)\times(2\times r+1)}(s, (x_i, x_{i+1}), r)$ to find out an element identical to d. Let us denote the found element as $S[p][q]$, where $p \in \{x_i - r + u \mid 0 \leqslant u \leqslant 2 \times r\}$, $q \in \{x_{i+1} - r + v \mid 0 \leqslant v \leqslant 2 \times r\}$, and $u \neq v$. Then, the stego pixel pair is computed by $(y_i, y_{i+1}) = (p, q)$. This embedding operation indicates that the proposed method conceals the secret number d into the cover pixel pair (x_i, x_{i+1}) by increasing or/and decreasing x_i or/and x_{i+1} at most by r or keeping x_i and x_{i+1} unmodified to obtain the stego pixel pair (y_i, y_{i+1}). Thus, the searching radius r can be regarded as the amplitude of the embedding distortion (i.e., the distortion caused by embedding) imposed on x_i and x_{i+1}. Especially, when $s = 2$ (and so $r = 1$), similar to the EMD method, only either x_i or x_{i+1} needs to be changed at most by 1 to obtain the stego pixel pair (y_i, y_{i+1}). The proposed embedding procedure is iteratively performed for the next cover pixel pair until all the secret numbers are concealed into the cover image X to obtain the stego image Y. The proposed method embeds $\lfloor log_2 s^2 \rfloor$ secret bits into each cover pixel pair so its hiding capacity is $\lfloor log_2 s^2 \rfloor / 2$ bpp.

It is noted that the found element $S[p][q]$ mentioned above may not be optimal in the sense of minimum embedding distortion. Therefore, the proposed method uses the minimum distortion embedding (MDE) process to find out an element $S[p][q]$ in the square $\boldsymbol{W}_{(2\times r+1)\times(2\times r+1)}(s, (x_i, x_{i+1}), r)$ that is identical to d and has a minimum embedding distortion. Specifically, when $s \geqslant 4$, there may be two or three found elements $S(x_a, y_a)$, $S(x_t, y_t)$, and $S(x_w, y_w)$ in the searching square $\boldsymbol{W}_{(2\times r+1)\times(2\times r+1)}(s, (x_i, x_{i+1}), r)$ that are equal to d. In this case, the element $S(x_j, y_j)$ with the minimum embedding distortion $d_{\min}$, which is defined by Eq. (4), is chosen.

$$D_{min} = min_{j=a,t,w}\{|x_i - x_j| + |x_i + 1 - y_j|\}. \quad (4)$$

Thus, the stego pixel pair is achieved by $(y_i, y_{i+1}) = (x_j, y_j)$. By this way, the proposed method can achieve the minimum embedding distortion caused by the embedding phase. For example, if the secret number $d = 10$ needs to be embedded into the cover pixel pair $(x_i, x_{i+1}) = (4, 3)$, then, as shown in Fig. 3, there are three elements $S[2][1]$, $S[2][5]$, and $S[6][2]$ that equals d. According to Eq. (4), the element $S[6][2]$ is selected so the stego pixel pair is $(y_i, y_{i+1}) = (6, 2)$. The embedding procedure of the proposed method is summarized as follows.

The proposed embedding procedure:

Input: The grayscale cover image X sized $H \times W$, the binary secret message $M = (m_1 m_2 \ldots m_{LM})$, the parameter s, where $2 \leqslant s \leqslant 23$.

Output: The grayscale stego image Y sized $H \times W$.

Step 1: Generate the matrix S sized 256×256 by using the proposed extraction function F defined by Eq. (2).
Step 2: Compute $k = \lfloor log_2 s^2 \rfloor$, $r = \lfloor s/2 \rfloor$.
Step 3: Set $i = 1$.
Step 4: Read the next k secret bits $(m_1 m_2 \ldots m_k)$ from M and convert them into the decimal number d.
Step 5: Read the next cover pixel pair (x_i, x_{i+1}) from X according to the user-defined pairing rule.
Step 6: If $d = S[x_i][x_{i+1}]$, then the grayscale stego pixel pair is attained by $(y_i, y_{i+1}) = (x_i, x_{i+1})$. Otherwise, search from the searching square $\boldsymbol{W}_{(2\times r+1)\times(2\times r+1)}(s, (x_i, x_{i+1}), r)$ to find out the element $S[p][q] = d$ with the minimum embedding distortion according to Eq. (4). Then, the grayscale stego pixel pair is achieved by $(y_i, y_{i+1}) = (p, q)$.

Step 7: Set $i = i + 2$.
Step 8: Repeat Steps 4–7 until all secret bits are embedded.

An example is now given to demonstrate the embedding process of the proposed method. Let us assume that $s = 4$, so $k = \lfloor log_2 s^2 \rfloor = 4$ and $r = \lfloor s/2 \rfloor = 2$. Suppose that four secret bits are $(m_1 m_2 m_3 m_4) = (1110)$. Thus, the secret number is $d = 14$. We now want to embed the secret number d into the cover pixel pair $(x_i, x_{i+1}) = (4, 3)$. As shown in Fig. 3, we have $S[4][3] = 8$. Because $d = 14 \neq S[4][3] = 8$, search from the searching square $\boldsymbol{W}_{5\times5}(4, (4, 3), 2)$ to find out an element equal to d. There are two found elements, namely $S[2][2]$ and $S[6][3]$. According to Eq. (4), the element $S[6][3]$ is chosen and the stego pixel pair is obtained by $(y_i, y_{i+1}) = (6, 3)$.

3.2. The extracting phase

At the receiving side, with the received stego image Y, an authorized receiver who knows the value s can calculate $k = \lfloor log_2 s^2 \rfloor$. Next, each embedded secret number is extracted from each stego pixel pair (y_i, y_{i+1}) in the stego image Y by $d = F(y_i, y_{i+1})$. Then, the extracted secret number d is converted back to k original secret bits $(m_1 m_2 \dots m_k)$ of the original secret message M. The extracting procedure is repeatedly executed for the next stego pixel pair (y_i, y_{i+1}) until all the secret numbers d's are extracted. The extracted secret bits $(m_1 m_2 \dots m_k)$ are collected to retrieve the original secret message M. The extracting procedure of the proposed method is summarized as follows.

The proposed extracting procedure:
Input: The grayscale stego image Y sized $H \times W$, the parameter s.
Output: The original binary secret message $M = (m_1 m_2 \dots m_{LM})$

Step 1: Compute $k = \lfloor log_2 s^2 \rfloor$.
Step 2: Set $i = 1$ and M is an empty message.
Step 3: Read the next stego pixel pair (y_i, y_{i+1}) according to the user-defined pairing rule used in the embedding phase.
Step 4: Extract the next embedded secret number by $d = F(y_i, y_{i+1})$.
Step 5: Convert d into k secret bits $(m_1 m_2 \dots m_k)$ which are appended to M.
Step 6: Set $i = i + 2$.
Step 7: Repeat Steps 3–6 until all secret bits are extracted.

The embedding example given in the above embedding process is now taken to illustrate the extracting process of the proposed method. The received stego pixel pair is $(y_i, y_{i+1}) = (6, 3)$. First, compute $k = \lfloor log_2 s^2 \rfloor = 4$, where $s = 4$. Next, the embedded secret number is easily extracted by $d = F(6, 3) = 14$. Then, d is converted back to four secret bits $(m_1 m_2 m_3 m_4) = (1110)$.

The execution time consumed by Step 6 in the embedding phase of the proposed method can be reduced as follows. As we can see from Fig. 3 that when $s \geqslant 4$, the searching square $\boldsymbol{W}_{(2\times r+1)\times(2\times r+1)}(s, (x_i, x_{i+1}), r)$ contains some smaller searching squares concentric to it. These smaller searching squares are defined as

$$SS_{(2\times z+1)\times(2\times z+1)}((x_i, x_{i+1}), z) = \{S(x_i - z + u, x_{i+1} - z + v) | 0 \leqslant u \leqslant 2\times z, 0 \leqslant v \leqslant 2\times z, u \neq v, 1 \leqslant z < r\}. \quad (5)$$

Thus, searching the element $S[p][q]$ equal to d in Step 6 of the embedding phase is started from the smallest searching square to the larger one until the element $S[p][q]$ is found. If the element $S[p][q]$ identical to d is found in the searching square smaller than the searching square $\boldsymbol{W}_{(2\times r+1)\times(2\times r+1)}(s, (x_i, x_{i+1}), r)$, then the searching process is stopped and the (MDE) process is not executed. As a result, the execution time becomes faster.

Searching from $\boldsymbol{W}_{(2\times r+1)\times(2\times r+1)}(s, (x_i, x_{i+1}), r)$ to find out an element identical to d does not work if x_i or x_{i+1} belongs to the extreme range $[0, r)$ or $(255 - r, 255]$. This is because the values of $x_i - r + u$ and $x_{i+1} - r + v$ may be less than 0 or greater than 255. There are some ways to deal with this problem. The simplest solution to this problem is as follows. The pixel values belonging to the extreme range $[0, r)$ are assigned to r. The pixel values belonging to extreme range $(255 - r, 255]$ are set to be $255 - r$. By this way, the proposed embedding procedure works correctly. Experimentally, for natural images, the number of pixels whose values belong to the extreme range $[0, r)$ or $(255 - r, 255]$ is very small. Thus, the impact of the above solution on the visual quality of stego images can be neglected.

4. Experimental results and discussion

To evaluate the performance of the proposed method, we implemented the LSB replacement (LSB-R) method Turner, 1989, Mielikainen's scheme (2006), Zhang and Wang's scheme (2006), Yang et al.'s scheme (2008), and the proposed scheme by using Borland C++ Builder 6.0 software running on the Pentium IV, 3.6 GHz CPU, and 1.49 GB RAM hardware platform. The binary secret message M of length LM was randomly generated by using the library function random() and used for the simulated methods. For simplicity, the pixels were processed in raster scan order to embed secret bits. The EMD method was implemented for the case of $n = 2$. Twelve commonly used grayscale images sized 512×512, as shown in Fig. 4, were used as the cover images in our simulations to test the performance of the proposed method in terms of hiding capacity and visual quality of stego images.

We used the peak signal-to-noise ratio (PSNR) Kutter and Petitcolas, 1999 to measure the distortion between the original cover image X and the stego image Y. The PSNR is defined by $PSNR = 10 \times \log_{10}(255^2/MSE)$ (dB), where MSE is the mean square error representing the distortion between the original cover image X sized $H \times W$ and the stego image Y sized $H \times W$. That is,

Fig. 4. Twelve grayscale test images sized 512×512.

Table 1
Performance results of Mielikainen's method, Zhang and Wang's method with $n = 2$ and the proposed method with $s = 2$.

Cover images	Mielikainen's method		EMD method ($n = 2$)		Proposed method ($s = 2$)	
	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR
Lena	1	52.39	1	52.09	1	52.39
Baboon	1	52.39	1	52.10	1	52.40
F16	1	52.39	1	52.12	1	52.38
Barbara	1	52.37	1	52.11	1	52.39
Boat	1	52.39	1	52.10	1	52.39
Goldhill	1	52.40	1	52.11	1	52.40
Elaine	1	52.39	1	52.10	1	52.39
Toys	1	52.38	1	52.11	1	52.39
Tiffany	1	52.38	1	52.11	1	52.39
Zelda	1	52.39	1	52.11	1	52.39
Pepper	1	52.39	1	52.11	1	52.38
Bridge	1	52.34	1	52.04	1	52.33
Average	1	52.38	1	52.1	1	52.39

Table 2
Performance results of the proposed method for various values of s.

Cover images	$s = 3$		$s = 4$		$s = 6$		$s = 8$	
	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR
(a) Performance results of the proposed method with s = 3, 4, 6, 8								
Lena	1.5	49.88	2	46.75	2.5	43.29	3	40.83
Baboon	1.5	49.89	2	46.74	2.5	43.29	3	40.82
F16	1.5	49.89	2	46.74	2.5	43.31	3	40.82
Barbara	1.5	49.89	2	46.75	2.5	43.30	3	40.84
Boat	1.5	49.90	2	46.76	2.5	43.30	3	40.82
Goldhill	1.5	49.89	2	46.76	2.5	43.31	3	40.83
Elaine	1.5	49.88	2	46.76	2.5	43.30	3	40.83
Toys	1.5	49.90	2	46.74	2.5	43.30	3	40.83
Tiffany	1.5	49.89	2	46.74	2.5	43.30	3	40.82
Zelda	1.5	49.89	2	46.75	2.5	43.30	3	40.82
Pepper	1.5	49.89	2	46.74	2.5	43.30	3	40.82
Bridge	1.5	49.86	2	46.68	2.5	43.21	3	40.75
Average	1.5	49.89	2	46.74	2.5	43.29	3	40.82

Cover images	$s = 12$		$s = 16$		$s = 23$	
	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR
(b) Performance results of the proposed method with s = 12, 16, 23						
Lena	3.5	37.31	4	34.83	4.5	31.70
Baboon	3.5	37.32	4	34.83	4.5	31.70
F16	3.5	37.32	4	34.82	4.5	31.70
Barbara	3.5	37.32	4	34.82	4.5	31.69
Boat	3.5	37.31	4	34.82	4.5	31.69
Goldhill	3.5	37.31	4	34.83	4.5	31.71
Elaine	3.5	37.32	4	34.83	4.5	31.71
Toys	3.5	37.33	4	34.82	4.5	31.70
Tiffany	3.5	37.32	4	34.84	4.5	31.71
Zelda	3.5	37.32	4	34.83	4.5	31.69
Pepper	3.5	37.33	4	34.83	4.5	31.66
Bridge	3.5	37.24	4	34.75	4.5	31.59
Average	3.5	37.31	4	34.82	4.5	31.69

$MSE = \frac{1}{H \times W}\sum_{i=1}^{H}\sum_{j=1}^{W}(X_{ij} - Y_{ij})^2$, where X_{ij} and Y_{ij} are the grayscale values of the pixels located at the ith row and the jth column of the cover image X and the stego image Y, respectively. Hiding capacity C (also called capacity for short) is measured by bit per pixel (bpp). That is, it is computed by the ratio between the total number of hidden secret bits and the total number of pixels in the cover image X.

The hiding capacities and the PSNR values of Mielikainen's method (2006), Zhang and Wang's method with $n = 2$ (Zhang & Wang, 2006), and the proposed method with $s = 2$ for test images are shown in Table 1.

Table 1 shows that the hiding capacity of Mielikainen's method, the EMD method with $n = 2$, and the proposed method with $s = 2$ is the same and equal to 1 bpp because in this case three schemes embed two secret bits into a cover pixel pair. It can be seen from Table 1 that the PSNR values of Mielikainen's method and the proposed method are nearly identical and greater than those of Zhang and Wang's method. This can be explained as follows. For the EMD method with $n = 2$, the probability that the secret digit $d \in \{0, 1, 2, 3, 4\}$ equals $R[x_i][x_{i+1}]$ is 0.200. Equivalently, the probability that d differs from $R[x_i][x_{i+1}]$ is 0.800. Thus, the expected number of modifications per pixel (ENMPP) of the EMD method is $\text{ENMPP}_{\text{EMD}} = 0.800/2 = 0.400$. According to the embedding rules of Mielikainen's method, the probability that $m_i = \text{LSB}(x_i)$ and $m_{i+1} = f(x_i, x_{i+1})$ (i.e., there is no modification in this case) is 0.250. Equivalently, the probability that either x_i or x_{i+1} has to be modified is 0.750. Therefore, the ENMPP of this method is $\text{ENMPP}_{\text{LSB-M-R}} = 0.750/2 = 0.375$. For the proposed method with $s = 2$, the probability that the secret number $d \in \{0, 1, 2, 3\}$ is equal to $S[x_i][x_{i+1}]$ is 0.250. Equivalently, the probability that d differs from $S[x_i][x_{i+1}]$ is 0.750.

Thus, the ENMMPP of the proposed method is $ENMPP_{Proposed} = 0.750/2 = 0.375 = ENMPP_{LSB\text{-}M\text{-}R}$. The above analyses explain the reason why the PSNR values of the proposed method are the same as those of the Mielikainen's scheme and greater than those of the EMD method.

The value of the parameter s in the proposed method is limited to be less than or equal to 23 because the visual quality of stego images produced by the proposed method is severely degraded (i.e., PSNR < 29 dB) when $s > 23$. The hiding capacities (i.e., $\lfloor log_2 s^2 \rfloor/2$ bpp for $2 \leqslant s \leqslant 23$) and the PSNR values of the proposed method with $s = 3, 4, 6, 8, 12, 16$, and 23 for test images are shown in Table 2. It can be observed from Table 2 that the proposed method can achieve various high hiding capacities of 1.5, 2, 2.5, 3, 3.5, 4, and 4.5 bpp for different values $s = 3, 4, 6, 8, 12, 16$, and 23, respectively, with good and acceptable visual qualities of stego images.

The stego images of Lena image produced by the proposed method for various hiding capacities are shown in Fig. 5 to verify the good and acceptable visual qualities of the proposed method. Fig. 5 shows that the visual quality of the stego Lena image embedded with 4.5 bpp is still acceptable and better than the LSB-R method embedded with 4 bpp, as shown in Fig. 6.

The performance of the proposed method is now compared with the recently published work. In late 2008, Yang et al. proposed an adaptive least significant bit replacement (A-LSB-R) steganographic method (Yang et al., 2008) using the pixel-value differencing (PVD) Wu & Tsai, 2003 and the minimum-error

Fig. 5. The stego Lena images embedded with various hiding capacities by the proposed method.

(a) 3 bpp, PSNR = 37.91 dB (b) 4 bpp, PSNR = 31.82 dB

Fig. 6. The stego Lena images embedded with 3 and 4 bpp by the LSB-R method.

Table 3
Performance results of Yang et al.'s method for various $l - h$ divisions with dividing line $D_{12} = 7$.

Cover images	2-3			2-4			3-4		
	Hidden bits	C	PSNR	Hidden bits	C	PSNR	Hidden bits	C	PSNR
Lena	575188	2.2	44.12	626088	2.4	39.80	837332	3.2	37.93
Baboon	651470	2.5	42.26	778652	3.0	36.69	913614	3.5	35.96
F16	563118	2.1	44.61	601948	2.3	40.75	825262	3.1	38.50
Barbara	628706	2.4	42.88	733124	2.8	37.67	890850	3.4	36.68
Boat	586816	2.2	43.83	649344	2.5	39.25	848960	3.2	37.65
Goldhill	600366	2.3	43.28	676444	2.6	38.28	862510	3.3	36.96
Elaine	621052	2.4	42.70	717816	2.7	37.31	883196	3.4	36.34
Toys	565678	2.2	44.41	607068	2.3	40.48	827822	3.2	38.27
Tiffany	566992	2.2	44.40	609696	2.3	40.34	829136	3.2	38.21
Zelda	573620	2.2	43.96	622952	2.4	39.56	835764	3.2	37.66
Pepper	561236	2.1	44.56	598184	2.3	40.73	823380	3.1	38.41
Bridge	667092	2.5	41.73	809896	3.1	36.03	929236	3.5	35.50
Average	596778	2.3	43.56	669268	2.6	38.91	858922	3.3	37.34

replacement (MER) technique Lee and Chen, 2000. Yang et al.'s scheme provides various hiding capacities and good visual qualities of stego images. Yang et al.'s method obeys the basic concept that the edge areas can tolerate more changes than smooth areas. For every two-pixel block (x_i, x_{i+1}), the difference value d between x_i and x_{i+1} is computed by $d = |x_i - x_{i+1}|$. The A-LSB-R method embeds k secret bits into each pixel of a cover pixel pair (x_i, x_{i+1}) at a time. The two-pixel blocks located in the edge areas (i.e., determined by $d = |x_i - x_{i+1}|$) are embedded by a k-bit LSB-R method with a larger value of k than that of the two-pixel blocks located in smooth areas. The details of Yang et al.'s method can be found in Yang et al. (2008).

The hidden bits (i.e., the total number of embedded secret bits), hiding capacities (denoted as C for short), and the PSNR values of Yang et al.'s method are shown in Tables 3–5. By observing the numerical data in Tables 2–5, the following comparisons are made to compare the performance of Yang et al.'s and proposed methods in terms of hiding capacity measured by bpp and visual quality measured by PSNR. First, the performance of the proposed scheme at the hiding capacity of 2.5 bpp (i.e., Table 2(a), $s = 6$) is compared with Yang et al.'s method for the 2-3 (i.e., Table 3) and 2-4 (i.e., Table 4(a)) divisions. As for the 2-3 division, it can be seen that the PSNR values of the two schemes are similar (i.e., around 43 dB) but the hiding capacity of Yang et al.'s method is less than 2.5 bpp whereas that of the proposed method is 2.5 bpp. Thus, in this case, the proposed method has better performance compared to Yang et al.'s method. With regard to the 2-4 division, it can be observed that the PSNR values (i.e., around 41.81 dB) and the hiding capacities (i.e., about 2.3 bpp) of Yang et al.'s method are less than those of the proposed method (i.e., 43.29 dB and 2.5 bpp, respectively). This confirms the superiority of the proposed method over Yang et al.'s method in this case.

Second, the performance of Yang et al.'s method for the 2-4 (i.e., Table 3), 2-5, and 3-4 (i.e., Table 4(a)) divisions is compared with the proposed scheme at the hiding capacity of 3 bpp (i.e., Table 2(a), $s = 8$). In regard to the 2-4 and 2-5 divisions, the numerical data show that the proposed method outperforms Yang et al.'s method. Specifically, Yang at el.'s method has the PSNR values less than 39 dB and hiding capacities smaller than 3 bpp whereas the proposed scheme achieves the PSNR values around 40.82 dB and hiding capacities equal to 3 bpp. With respect to the 3-4 division, the performances of Yang et al.'s and proposed methods are marginally comparable. That is, in average, the hiding capacity and the PSNR value of Yang et al.'s method are 3.1 bpp and 39.04 dB, respectively, and those of the proposed method are 3 bpp and 40.82 dB, respectively.

Third, the performance of the proposed scheme at the hiding capacity of 3.5 bpp (i.e., Table 2(b), $s = 12$) is compared with Yang et al.'s method for the 3-4 (i.e., Table 3), 3-5 (i.e., Table 4(b)), 3-4-5, and 3-4-6 (i.e., Table 5) divisions. As for the 3-5 and 3-4-6 divisions, in average, Yang et al.'s method obtains the PSNR value less than 36 dB and the hiding capacity smaller than 3.5 bpp, respectively. In contrast, the proposed method can offer the hiding capacity of 3.5 bpp and the PSNR value of 37.31 dB. These results demonstrate that the proposed method surpasses Yang et al.'s method. In respect of the 3-4 and 3-4-5 divisions, the PSNR values of Yang et al.'s scheme are comparable to those of the proposed method. That is, both schemes can achieve the PSNR value of about

Table 4
Performance results of Yang et al.'s method for various *l-h* divisions 2-4, 2-5, 3-4, 3-5, 4-5, and 4-6 with dividing line $D_{12} = 15$.

Coverimages	2-4			2-5			3-4		
	Hidden bits	C	PSNR	Hidden bits	C	PSNR	Hidden bits	C	PSNR
(a) Performance results of Yang et al.'s method for various l – h divisions 2-4, 2-5, and 3-4 *with dividing line* $D_{12} = 15$									
Lena	565936	2.2	42.65	586760	2.2	37.70	807256	3.1	39.50
Baboon	654652	2.5	39.24	719834	2.7	33.20	851614	3.2	37.74
F16	561532	2.1	43.01	580154	2.2	38.31	805054	3.1	39.67
Barbara	662640	2.5	39.37	731816	2.8	33.53	855608	3.3	37.89
Boat	587580	2.2	41.67	619226	2.4	36.35	818078	3.1	39.05
Goldhill	578272	2.2	41.90	605264	2.3	36.66	813424	3.1	39.14
Elaine	579792	2.2	41.56	607544	2.3	36.17	814184	3.1	38.88
Toys	561392	2.1	43.10	579944	2.2	38.48	804984	3.1	39.71
Tiffany	556808	2.1	43.22	573068	2.2	38.64	802692	3.1	39.73
Zelda	546912	2.1	43.87	558224	2.1	39.75	797744	3.0	39.96
Pepper	551760	2.1	43.59	565496	2.2	39.22	800168	3.1	39.89
Bridge	677840	2.6	38.50	754616	2.9	32.47	863208	3.3	37.26
Average	590426	2.3	41.81	623496	2.4	36.71	819501	3.1	39.04
Coverimages	3-5			4-5			4-6		
	Hidden bits	C	PSNR	Hidden bits	C	PSNR	Hidden bits	C	PSNR
(b) Performance results of Yang et al.'s method for various l-h divisions 3-5, 4-5, and 4-6 *with dividing line* $D_{12} = 15$									
Lena	828080	3.2	36.42	1069400	4.1	33.28	1090224	4.2	30.18
Baboon	916796	3.5	32.81	1113758	4.2	31.27	1178940	4.5	26.42
F16	823676	3.1	36.88	1067198	4.1	33.50	1085820	4.1	30.62
Barbara	924784	3.5	33.08	1117752	4.3	31.60	1186928	4.5	26.72
Boat	849724	3.2	35.44	1080222	4.1	32.83	1111868	4.2	29.12
Goldhill	840416	3.2	35.61	1075568	4.1	32.86	1102560	4.2	29.31
Elaine	841936	3.2	35.22	1076328	4.1	32.56	1104080	4.2	28.98
Toys	823536	3.1	37.00	1067128	4.1	33.55	1085680	4.1	30.82
Tiffany	818952	3.1	37.08	1064836	4.1	33.53	1081096	4.1	30.63
Zelda	809056	3.1	37.77	1059888	4.0	33.74	1071200	4.1	31.59
Pepper	813904	3.1	37.54	1062312	4.1	33.71	1076048	4.1	31.32
Bridge	939984	3.6	32.13	1125352	4.3	30.89	1202128	4.6	25.75
Average	852570	3.3	35.58	1081645	4.1	32.78	1114714	4.3	29.29

Table 5
Performance results of Yang et al.'s method for various *l-m-h* divisions with dividing line $D_{12} = 15$ and $D_{23} = 31$.

Cover images	3-4-5			3-4-6		
	Hidden bits	C	PSNR	Hidden bits	C	PSNR
Lena	812794	3.1	38.32	818332	3.1	35.53
Baboon	870790	3.3	35.34	889966	3.4	31.09
F16	812412	3.1	38.25	819770	3.1	34.96
Barbara	894600	3.4	34.41	933592	3.6	29.23
Boat	831260	3.2	37.07	844442	3.2	33.05
Goldhill	820288	3.1	37.83	827152	3.2	34.81
Elaine	816956	3.1	38.31	819728	3.1	36.65
Toys	813426	3.1	38.10	821868	3.1	34.74
Tiffany	807682	3.1	38.71	812672	3.1	35.76
Zelda	799828	3.1	39.41	801912	3.1	37.72
Pepper	804266	3.1	38.96	808364	3.1	36.60
Bridge	889760	3.4	34.60	916312	3.5	30.00
Average	831172	3.2	37.44	842843	3.2	34.18

37.3 dB. However, in these cases, the hiding capacity of the proposed method is of 3.5 bpp whereas that of Yang et al.'s method is around 3.3 bpp. Thus, it can be said that in these cases, the performance of Yang et al.'s method is inferior to that of the proposed method.

Fourth, the performance of Yang et al.'s method for the 4-5 division (i.e., Table 4(b)) is compared with the proposed scheme at the hiding capacity of 4 bpp (i.e., Table 2(b), $s = 16$). It can be seen that Yang et al.'s method can achieve higher hiding capacity (i.e., around 4.1 bpp) at the cost of lower PSNR value (i.e., about 32.78 dB). In this case, the hiding capacity and the PSNR value of the proposed method are 4 bpp and 34.82 dB, respectively. Thus, in this case, the performances of the two schemes are comparable. Finally, the performance of the proposed scheme at the hiding capacity of 4.5 bpp (i.e., Table 2(b), $s = 23$) is compared with Yang et al.'s method for the 4-6 division (i.e., Table 4(b)). In this case, the proposed scheme can obtain the hiding capacity of 4.5 bpp and the PSNR value of at least 31.59 dB whereas Yang et al.'s method achieves the hiding capacity of around 4.3 bpp and the PSNR value of at most 31.59 dB. Thus, it is inferred that the proposed method has a better performance compared to Yang et al.'s method.

5. Conclusions

In this paper, we propose a novel steganographic scheme by fully exploiting the modification directions. The merits of this paper are summarized as follows. First, the novel extraction function is proposed to devise an efficient steganographic scheme. Secondly, the proposed method improves the visual quality (i.e., the ENMMP value) of Zhang and Wang's method at hiding capacity of 1 bpp. Thirdly, the ENMPP value of the proposed approach is equal to that of the Mielikainen's method at hiding capacity of 1 bpp. Fourthly, by utilizing eight modification directions, restricting the embedding distortion into the searching square of various sizes, and using the minimum distortion embedding (MDE) process for the embedding phase, the proposed method can provide various hiding capacities (i.e., $\lfloor log_2 s^2 \rfloor/2$ bpp for $2 \leqslant s \leqslant 23$) with good and acceptable visual qualities to satisfy different requirements of users. Thus, we can conclude that the proposed method has some merits and is applicable to steganographic applications such as convert communications.

References

Chan, C. K., & Cheng, L. M. (2004). Hiding data in images by simple LSB substitution. *Pattern Recognition, 37*(3), 469–474.

Chang, C. C., Hsiao, J. Y., & Chan, C. S. (2003). Finding optimal least-significant-bit substitution in image hiding by dynamic programming. *Pattern Recognition, 36*(7), 1583–1595.

Chang, C. C., Kieu, T. D., & Chou, Y. C. (2009a). Reversible information hiding for VQ indices based on locally adaptive coding. *Journal of Visual Communication and Image Representation, 20*(1), 57–64.

Chang, C. C., Kieu, T. D., & Wu, W. C. (2009). A lossless data embedding technique by joint neighboring coding. *Pattern Recognition, 42*(7), 1597–1603.

Chang, C. C., Tai, W. L., & Lin, C. C. (2006). A reversible data hiding scheme based on side match vector quantization. *IEEE Transactions on Circuits and Systems for Video Technology, 16*(10), 1301–1308.

Cox, I. J., Miller, M. L., Bloom, J. A., Fridrich, J., & Kalker, T. (2007). *Digital watermarking and steganography*. Morgan Kaufmann. ISBN: 978-0-12-372585-1.

Davis, R. M. (1978). The data encryption standard in perspective. *IEEE Communications Magazine, 16*(6), 5–9.

Fridrich, J., Goljan, M., & Du, R. (2001). Detecting steganography in color and gray-scale images. *IEEE Multimedia, 8*(4), 22–28.

Fridrich, J., Goljian, M., & Soukal, D. (2006). Wet paper codes with improved embedding efficiency. *IEEE Transactions on Information Forensics and Security, 1*(1), 102–110.

Fridrich, J., Lisonek, P., & Soukal, D. (2007). On steganographic embedding efficiency. In *Proceedings of 8th international workshop on information hiding. LNCS* (vol. 4437, pp. 282–296). Springer.

Harmsen, J., & Pearlman, W. (2003). Steganalysis of additive-noise modelable information hiding. In *Proceedings of SPIE: Security and watermarking of multimedia contents* (Vol. 5020, pp. 131–142).

Ker, A. D. (2004). Improved detection of LSB steganography in grayscale images. In *Proceedings of 6th international workshop on information hiding. LNCS* (Vol. 3200, pp. 97–115). Springer.

Ker, A. D. (2005). Steganalysis of LSB matching in grayscale images. *IEEE Signal Processing Letters, 12*(6), 441–444.

Kutter, M., & Petitcolas, F. A. P. (1999). A fair benchmark for image watermarking systems. In *Proceedings of SPIE: Security and watermarking of multimedia contents* (Vol. 3657, pp. 226–239).

Lan, T. H., & Tewfik, A. H. (2006). A novel high-capacity data-embedding system. *IEEE Transactions on Image Processing, 15*(8), 2431–2440.

Langelaar, G. C., Setyawan, I., & Lagendijk, R. L. (2000). Watermarking digital image and video data: a state-of-the-art overview. *IEEE Signal Processing Magazine, 17*(5), 20–46.

Lee, Y. K. & Chen, L. H. (2000). High capacity image steganographic model. In *IEE proceedings of vision, image, and signal processing* (Vol. 147, pp. 288–294).

Lee, S., Yoo, C. D., & Kalker, T. (2007). Reversible image watermarking based on integer-to-integer wavelet transform. *IEEE Transactions on Information Forensics and Security, 2*(3), 321–330.

Mielikainen, J. (2006). LSB matching revisited. *IEEE Signal Processing Letters, 13*(5), 285–287.

Ni, Z., Shi, Y., Ansari, N., & Su, W. (2006). Reversible data hiding. *IEEE Transactions on Circuits and Systems for Video Technology, 16*(3), 354–362.

Ni, Z., Shi, Y. Q., Ansari, N., Su, W., Sun, Q., & Lin, X. (2008). Robust lossless image data hiding designed for semi-fragile image authentication. *IEEE Transactions on Circuits and Systems for Video Technology, 18*(4), 497–509.

Petitcolas, F. A. P., Anderson, R. J., & Kuhn, M. G. (1999). Information hiding – A survey. *Proceedings of the IEEE, 87*(7), 1062–1078.

Rivest, R., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM, 21*(2), 120–126.

Sharp, T. (2001). An implementation of key-based digital signal steganography. In *Proceedings of 4th international workshop on information hiding. LNCS* (Vol. 2137, pp. 13–26). Springer.

Turner, L. F. (1989). *Digital data security system*. Patent IPN, WO 89/08915.

Wang, R. Z., Lin, C. F., & Lin, J. C. (2001). Image hiding by optimal LSB substitution and genetic algorithm. *Pattern Recognition, 34*(3), 671–683.

Westfeld, A. (2001). F5-A steganographic algorithm. In *Proceedings of 4th international workshop on information hiding. LNCS* (Vol. 2137, pp. 289–302). Springer.

Wu, H. C., Wu, N. I., Tsai, C. S., & Hwang, M. S. (2005). Image steganographic scheme based on pixel-value differencing and LSB replacement methods. In *IEE proceedings of vision, image, and signal processing* (Vol. 152, pp. 611–615).

Wu, D. C., & Tsai, W. H. (2003). A steganographic method for images by pixel-value differencing. *Pattern Recognition Letters, 24*(9–10), 1613–1626.

Yang, C. H., Weng, C. Y., Wang, S. J., & Sun, H. M. (2008). Adaptive data hiding in edge areas of images with spatial LSB domain systems. *IEEE Transactions on Information Forensics and Security, 3*(3), 488–497.

Zhang, X., & Wang, S. (2006). Efficient steganographic embedding by exploiting modification direction. *IEEE Communications Letters, 10*(11), 1–3.