Journal of Computers Vol. 28, No. 1, 2017, pp. 13-29
doi:10.3966/199115592017022801002

A Steganographic Method by Pixel-Value Differencing and Exploiting Modification Direction

Chwei-Shyong Tsai[1], Hsien-Chu Wu[2], Chih-Chiang Lee[1],
Tsung-Fu Shie[1], and Cheng-Chi Lee[3,4*]

[1] Department of Management Information Systems, National Chung Hsing University
Taichung, Taiwan, R.O.C.
tsaics@nchu.edu.tw

[2] Department of Computer Science and Information Engineering, National Taichung University of Science and Technology
Taichung, Taiwan, R.O.C.
wuhc@nutc.edu.tw

[3] Department of Library and Information Science, Fu Jen Catholic University
New Taipei City, Taiwan, R.O.C.

[4] Department of Photonics and Communication Engineering, Asia University
Taichung, Taiwan, R.O.C.
*Corresponding: cclee@mail.fju.edu.tw

Received 8 June 2015; Revised 22 July 2015; Accepted 26 January 2016

Abstract. In this paper, we shall propose a new image steganographic scheme for embedding secret messages into a gray-valued cover image and capable of producing a secret-embedded image that is totally indistinguishable from the original image by the human eye. In addition, the proposed scheme avoids the falling-off-boundary problem by using the Pixel-Value Differencing (PVD) and the Exploiting Modification Direction (EMD). A cover image is partitioned into non-overlapping blocks of four consecutive pixels. First, we derive two pieces of difference value from four consecutive pixels by utilizing the PVD. The hiding capacity of the four consecutive pixels depends on the two pieces of difference value. In other words, the smoother area is, the less secret data can be hidden; on the contrary, the more edges an area has, the more secret data can be embedded, and then secret data embedded by using the PVD. In order to provide high payload capacity, the remainder of two average values of the four consecutive pixels can be computed, and the two average values are capable of embedding secret data by using the EMD. The embedded secret message can be extracted from the resulting stego-image without referencing the original cover image. Moreover, a pseudo-random mechanism may be used to achieve secrecy protection. The experimental results show that the proposed scheme is capable of providing a great payload capacity, and the image quality of the embedded image is better than PVD for a gray-level image.

Keywords: data embedding, dual layer, exploiting modification direction, pixel-value differencing, steganography

[*] Corresponding Author

1 Introduction

Since the development of Internet, the Internet technology leads people into the wholly different world than before, and it is very convenient and frequent to communicate with other people via Internet. However, new issues also arose and have been noticed [1, 2], such as data security in public channel communications, copyright protection of digitized properties, invisible communication via digital media, personal private matter protections on Internet, etc.

Information/data hiding is a commonly used technique that embeds additional message into the host signals such that an unintended observer will not be aware of the existence of the hidden messages. These messages can serve as authentication codes, annotation, or secret data depending on the purpose of the application itself. In recent years, enormous research efforts have been invested in the development of digital image steganographic techniques [3-11]. The major goal of steganography is to enhance communication security by inserting secret message into the digital image without any suspiciousness. For the purpose of increasing the embedding capacity and enhancing security, the secret message may be compressed and encrypted before the embedding steps begin. The image after the embedding of the secret message, so-called stego-image, is then sent to the receiver through a public channel. Digital images may use as the carriers of secret messages in steganographic methods to deliver or hide data like secret letters, military maps, favorite pictures, etc. In such data embedding applications with emphasizing on cheating or hiding, attackers do not know that the stego-image has included secret message, so they will not disturb the stego-image. Then in the transmission process, the public channel may be intentionally monitored by some opponent who tries to prevent the message from being successfully sent and received. The opponent may randomly attack the stego-image if he/she doubts the stego-image carries any secret message because the appearance of the stego-image shows obvious artifacts of hiding effect [12, 13]. For this reason, an ideal steganography scheme, to keep the stego-image from drawing attention from the opponent, should maintain an imperceptible stego-image quality. That is to say, if there are more similarities between the cover image and the stego-image, it will be harder for an attacker to find out that the stego-image has important secret data hidden inside it [14]. This way, the secret data is more likely to travel from the sender to the receiver safe and sound.

For the past decade, many steganographic techniques about embedding data in images have been proposed [14-23]. One of the common techniques is based on manipulating the Least-Significant-Bit (LSB) [16] planes by directly replacing the LSBs of the cover image with the message bits. LSB methods typically achieve high quality. Wang et al. proposed [18] a new scheme to embed secret messages in the moderately significant bit of the cover-image. A genetic algorithm is developed to find an optimal substitution matrix for the embedding of the secret messages. In order to improve the image quality of the stego-image, they also proposed to use a Local Pixel Adjustment Process (LPAP). Unfortunately, since the local pixel adjustment process only take the last three least significant bits and the fourth bit into consideration but not on all bits, in some cases the local pixel adjustment process is obviously not optimal. The weakness of the local pixel adjustment process is pointed out in [15]. As the local pixel adjustment process modifies the LSBs, the technique cannot be applied to data hiding schemes based on simple LSB substitution. Consequently, a genetic algorithm of optimal LSB substitution is also proposed by Wang et al. [19]. Using the proposed algorithm, the Worst Mean-Square-Error (WMSE) between the cover image and the stego-image is shown to be a half of that obtained by the simple LSB substitution method [17]. However, the LSB-based methods mentioned above, directly embed the secret data into the spatial domain in an unreasonable way without taking into consideration the difference in hiding capacity between edge and smooth areas. For human vision's sensitivity in general, the alteration tolerance of an edge area is higher than that of a smooth area. That is to say, an edge area can conceal more secret data than a smooth area. With this concept in mind, Wu and Tsai [20] presented a steganographic scheme that offers high imperceptibility to the stego-image by selecting two consecutive pixels as the object of embedding. The payload of Wu and Tsai's scheme is determined by the difference value between the pixels. Therefore, their scheme produces stego-images that are more similar to the original images than those produced by LSB substitution schemes.

In this paper, in order to provide both higher payload capacity and a better stego-image quality than Wu and Tsai's scheme [20], we shall propose a novel technique based on Pixel-Value Difference (PVD) and Exploiting Modification Direction (EMD). In Wu and Tsai's scheme, which is also known as the

PVD method, the difference value between two consecutive pixels is regarded as a feature for recording the secret message. When the original difference value is unequal to the secret message, the two consecutive pixels will be directly adjusted so that their difference value can stand for the secret data. However, considerable the remainder of average value of four consecutive pixels, which we can reuse to embedding more secret data. To make a difference, with our new method, we shall improve the payload capacity by reuse the remainder of average values of non-overlapping blocks, instead of embedding secret message by PVD at a time. This method provides an easy way to produce more imperceptible results than those yielded by simple LSB substitution methods. The method was designed in such a way that there is no need of using the original cover image in recovering the secret message from the stego-image. Besides that, we also take the falling-off-boundary problem into consideration. To overcome the falling-off-boundary problem, our new scheme re-revises the remainder of the average value of a block. Moreover, while hiding data in the cover image is traversed in an order provided by a pseudo-random number generator to achieve secrecy, and so to prevent tampering access to the embedded data from illicit users. Compare with other schemes, we also prove that our proposed scheme has better performance regarding to the payloads and PSNR values.

The remainder of this paper is organized as follows. In Section 2, we shall briefly review PVD and EMD. Then, in Section 3, we shall present the proposed scheme in detail. In section 4 the performance of the proposed scheme will be analyzed and compared with those of the existing schemes in terms of payload and stego-image quality. Finally, a brief conclusion will be given in Section 5.

2 Related Works

In this section, two data hiding techniques are to be briefly introduced. First one is PVD proposed by Wu and Tsai [20], based on embedding secret data into two consecutive pixels. The second one hides secret data by Exploiting Modification Direction (EMD), proposed by Zhang and Wang [21]. The details of the two schemes are in [20] and [21]. Please refer them.

2.1 PVD

Given a gray-value cover image F sized $M\times N$. The cover image F is partitioned into two-pixel blocks by runs through all the rows of cover image in a raster scan order such that $F=\left\{F_i \mid i=1,2,...,\frac{M\times N}{2}\right\}$. By definition each F_i contains two consecutive pixels, say $P_{(i,L)}$ and $P_{(i,R)}$. Assume that the gray values of $P_{(i,L)}$ and $P_{(i,R)}$ are $P_{(i,x)}$ and $P_{(i,y)}$ respectively. The difference value of $P_{(i,x)}$ and $P_{(i,y)}$ can derived by Eq. (1).

$$d_i = P_{(i,y)} - P_{(i,x)} \tag{1}$$

They also design a range table R which consists of n contiguous sub-ranges. In other words, $R=\{R_j \mid j=1,2,...,n\}$. The purpose of the range table is to provide information about the hiding capacity of each F_i. The lower and upper bound values of R_j are denoted by l_j and u_j respectively, where l_1 is 0 and u_n is 255. The width w_j of each R_j is selected to be a power of 2, and can be computed by $w_j = u_j - l_j + 1$. For example, the range widths of 8, 8, 16, 32, 64, and 128, which partition the total range of [0,255] into [0,7],[8,15],[16,31],…,[128,255]. The hiding capacity of two consecutive pixels can be obtained by Eq. (2).

$$b_i = \lfloor \log_2(w_j) \rfloor \tag{2}$$

Here, b_i is the number of secret bits that can be hidden in F_i. Read b_i bits from the binary secret data stream and transform b_i into its decimal value $b_i^{'}$. A new difference value $d_i^{'}$ can be generated by Eq. (3).

$$d_i^{'} = \begin{cases} b_i^{'} + l_j & for\ d_i \ge 0 \\ -(b_i^{'} + l_j) & for\ d_i < 0 \end{cases} \tag{3}$$

Now the secret data can be embedded into F_i by modifying its $P_{(i,x)}$ and $P_{(i,y)}$. The details of the embedding criteria are as follows:

$$(P'_{(i,x)},P'_{(i,y)})=\begin{cases}(P_{(i,x)}+\left\lceil \frac{m}{2}\right\rceil,P_{(i,y)}-\left\lfloor \frac{m}{2}\right\rfloor) \\ if\ d_i \bmod 2=1; \\ (P_{(i,x)}-\left\lfloor \frac{m}{2}\right\rfloor,P_{(i,y)}+\left\lceil \frac{m}{2}\right\rceil) \\ if\ d_i \bmod 2=0; \end{cases} \quad \textbf{(4)}$$

where $m=\left|d'_i-d_i\right|$. We can obtain new pixel values $P'_{(i,x)}$ and $P'_{(i,y)}$ after the calculation in Eq. (4) and then replace $P_{(i,x)}$ and $P_{(i,y)}$ in the cover image with the new pixel values so that the embedding process is accomplished. An illustration of how embedding secret data into cover image by adjusting $P_{(i,x)}$ and $P_{(i,y)}$ of Wu and Tsai's scheme is shown in Table 1, assume the pixel values of two consecutive pixels are $P_{(i,x)}=15$, $P_{(i,y)}=27$, and the width w_j of their sub-range is 16 with $l_j=8$ and $u_j=15$. Then, 3 bits of the secret data can be embedded into this block. All the possible ways of adjusting $P_{(i,x)}$ and $P_{(i,y)}$ are shown in Table 1.

Table 1. An illustration of Wu and Tsai's embedding process

Secret data (decimal value)	$P_{(i,x)}=15$	$P_{(i,y)}=27$	New difference value of $P_{(i,x)}$ and $P_{(i,y)}$
0	17	25	8
1	17	26	9
2	16	26	10
3	16	27	11
4	15	27	12
5	15	28	13
6	14	28	14
7	14	29	15

However, in some case, $(P_{(i,x)},P_{(i,y)})$ may produce invalid $(P'_{(i,x)},P'_{(i,y)})$, some of the calculation may cause the resulting $P'_{(i,x)}$ or $P'_{(i,y)}$ to fall off the boundaries of the range [0,255]. To deal with this case, using falling-off boundary checking proceeds by producing a pair of $(\hat{P}_{(i,x)},\hat{P}_{(i,y)})$ by Eq. (5).

$$(\hat{P}_{(i,x)},\hat{P}_{(i,y)})=f((P_{(i,x)},P_{(i,y)}),u_k-d_i)=\begin{cases}(P_{(i,x)}-\left\lceil \frac{u_k-d_i}{2}\right\rceil,P_{(i,y)}+\left\lfloor \frac{u_k-d_i}{2}\right\rfloor) \\ if\ d_i \bmod\ 2=1; \\ (P_{(i,x)}-\left\lfloor \frac{u_k-d_i}{2}\right\rfloor,P_{(i,y)}+\left\lceil \frac{u_k-d_i}{2}\right\rceil) \\ if\ d_i \bmod\ 2=0; \end{cases} \quad \textbf{(5)}$$

If either $P'_{(i,x)}$ or $P'_{(i,y)}$ falls off the boundary of 0 or 255, we regard the block to have the possibility of falling-off, and abandon the block for embedding data. The recovery process of Wu and Tsai's method is quite simple and easy. Given two consecutive pixels $P'_{(i,x)}$ and $P'_{(i,y)}$ of the stego-image, compute their difference value d'_i and obtain $d'_i=\left|P_{(i,y)}-P_{(i,x)}\right|$. Use the original range table R in the embedding phase to obtain the same R_j and w_j. The length b_i of the hiding capacity also can be gained by Eq. (2). Calculate decimal value $b_i^{'} = d_i^{'} - l_j$ of secret data by Eq. (6), and then covert the decimal value into a binary string whose length is b_i bits. The recovery process of the above example is shown in Table 2.

$$b_i^{'} = \begin{cases} d_i^{'} - l_j & \text{for } d_i^{'} \geq 0 \\ -d_i^{'} - l_j & \text{for } d_i^{'} < 0 \end{cases} \tag{6}$$

Table 2. An illustration of Wu and Tsai's recovery process

$P_{(i,x)}^{'}$	$P_{(i,y)}^{'}$	Difference value of $P_{(i,x)}^{'}$ and $P_{(i,y)}^{'}$	Decimal value of secret data ($b_i^{'} = d_i^{'} - l_j$)	Secret data string
17	25	8	0	0
17	26	9	1	1
16	26	10	2	10
16	27	11	3	11
15	27	12	4	100
15	28	13	5	101
14	28	14	6	110
14	29	15	7	111

2.2 EMD

The main idea of this method is that each secret digit in a (2*c*+1)-ary notational system is carried by *c* cover pixels and, at most, only one pixel is increased or decreased by 1. Before begin data embedding, first convert the secret message into a sequence of digits in the notational system with an odd base (2*c*+1). If the secret message is a binary stream, it can be segmented into many pieces with *L* bits, and the decimal value of each secret piece is represented by *K* digits in a (2*c*+1)-ary notational system, where

$$L = \lfloor K \cdot \log_2(2c+1) \rfloor \tag{7}$$

For example, the sequence of binary secret message (1110 0101 1010) can be covert as (24 10 20) in a 5-ary notational system where *L*=4 and *K*=2.

Now, if we want to embed data into a cover image, with this method. First, *c* pixels are used to conceal one secret digit in the (2*c*+1)-ary notational system. Besides, for the purpose of security, pseudo-randomly permute all cover pixels according to a secret key, and divide them into a series of pixel-groups, each containing *c* pixels. Then denote the gray values of pixels in a group as $g_1, g_2, ..., g_c$, finally calculate the extraction function *f* as a weighted sum modulo (2*c*+1) by Eq. (8).

$$f(g_1, g_2, ..., g_c) = \left[\sum_{i=1}^{c} (g_i \cdot i) \right] \bmod (2c+1) \tag{8}$$

No modification is needed if a secret digit d equals the extraction function of the original pixel-group. When $d \neq f$, calculate $s = d - f \bmod (2c+1)$ if s is no more than *c*, increase the value of g_s by 1, otherwise, decrease the value of g_{2c+1-s} by 1.

However, increase of g_s or decrease of g_{2c+1-s} may occur overflow problem. To deal with this problem, increases or decreases the saturated pixel by 1 and embed the secret digit again until the $d = f$. In the recovery process of this method is quite simple and easy, the secret digit is extracted by calculating the extraction function of steog-pixel-group. For example, assume the pixel group is (15, 27), and the secret data can extract by Eq. (8). So secret data $s = f(15, 27) \bmod 5 = 4$.

3 The Proposed Scheme

In this section, we shall present the proposed scheme, instead of embedding secret data into cover image using pixel value difference technique or exploiting modification direction method independently, on the contrary, our proposed scheme is hybrid from pixel value difference technique (PVD) and exploiting modification direction technique (EMD). This is an improved version of those two techniques where the distortion of the stego-image is reduced. The proposed scheme is capable of providing a great payload capacity, and the image quality of the embedded image is better than PVD for a gray-level image. To begin with, we partition cover image into a series of non-overlapping blocks, each block is composed of four consecutive pixels P_1, P_2, P_3, P_4, then we can select each sub-image by from top-down and left-right in turn for data embedding process. We can obtain two pixel value differences d_1, d_2 and two average numbers a_1, a_2 from four pixels of each block, where $d_1 = P_2 - P_1$, $d_2 = P_4 - P_3$ and $a_1 = \left\lfloor \frac{P_1 + P_2}{2} \right\rfloor$, $a_2 = \left\lfloor \frac{P_3 + P_4}{2} \right\rfloor$. Further, we use d_1, d_2 to embed secret data by pixel value difference technique, and use a_1, a_2 to embed secret data by exploiting modification direction technique. The payload of each block is variable and depends on the image complexity of the block. In order to avoid critical destruction, some blocks, such as those located in extreme edge areas, are not used for hiding data. The embedding and extracting procedures will be illustrated by the block diagrams shown in Fig. 1 and Fig. 2, respectively. Now, let's see how we embed secret data by our proposed scheme.

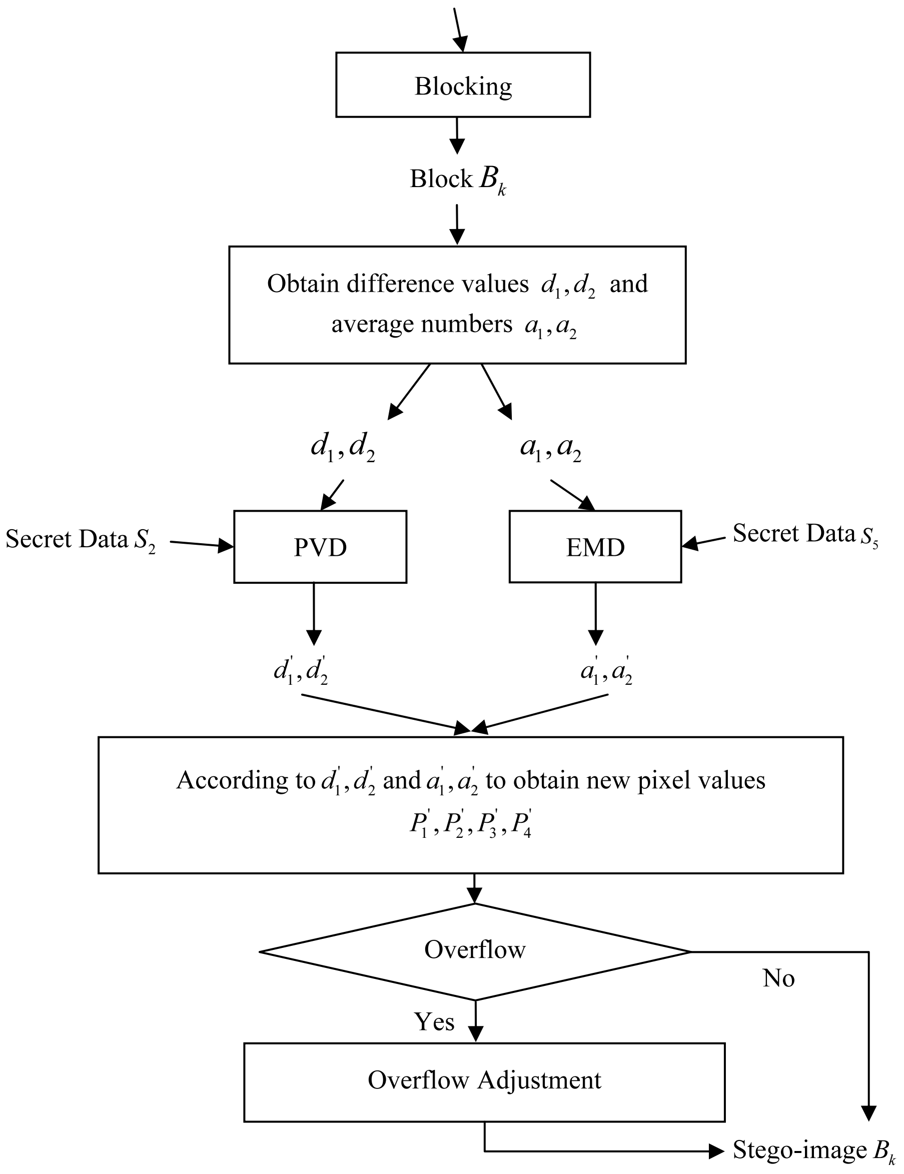


Fig. 1. Data embedding algorithm

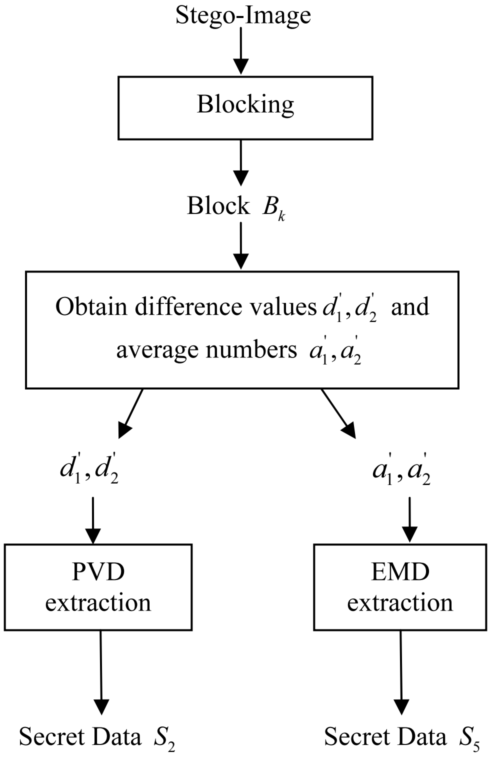


Fig. 2. Data extraction algorithm

3.1 Embedding Procedure—PVD

As defined above, for each component of a block, we can obtain two pixel value differences, d_1 and d_2 . Then according to d_1 and d_2 to find the hiding capacity from range table R_j , the width $w_j = u_j - l_j + 1$., the hiding capacity b_i bits is calculate by Eq. (2), individually. Suppose the secret data $S_2^i = S_2^1, S_2^2, S_2^3, ..., S_2^m$,$for\ i = 1,2,3,...m$, $\forall S_2^i \in [0,1]$. After that read b_i bits from S_2 and transform b_i bits of binary secret data into decimal values $b_i^{'}$ and $b_{i+1}^{'}$ for d_1 and d_2. Therefore, we can obtain two new pixel value differences $d_1^{'}$ and $d_2^{'}$ by using Eq. (3) of Wu and Tsai's scheme according to Section 2. An illustration of the data embedding process is shown in Fig. 3. In the figure, the gray values P_1 and P_2 of a sample are assumed to be (15,27). The difference value $d_1 = 12$, which is in the range of 8 through 15. The width of the range is $15 - 8 + 1 = 8$, which means that a difference value in the range can be used to embed for three bits of secret data. Suppose that the three leading bits of the secret data are 101. The decimal value of this secret bit stream is 5. It is added to the lower bound value 8 of the range to yield the new difference value $d_1^{'} = 8 + 5 = 13$. In the same way, after embedding secret data into the the P_3 and P_4, which yield the new difference value $d_2^{'} = -(96 + 20) = -116$.

3.2 Embedding Procedure—EMD

For each component of a block, we can obtain two average numbers a_1 and a_2, then we use a_1 and a_2 embedding secret data by EMD, as define on Section 2.1. Before begin embedding process, we need transform the binary secret data stream into 5-ary notational system. At first, we pick up 8 bits sequentially of the binary secret data stream once a time. Then we figure out the decimal value of the 8 bits, after that, transform the decimal value into 5-ary notational system. Suppose after transformation the secret data is $S_5^j = S_5^1 S_5^2 S_5^3 ... S_5^n, for\ j = 1,2,3,...n$,$\forall S_5^j \in [0,1,2,3,4]$. Now we can start to embed secret

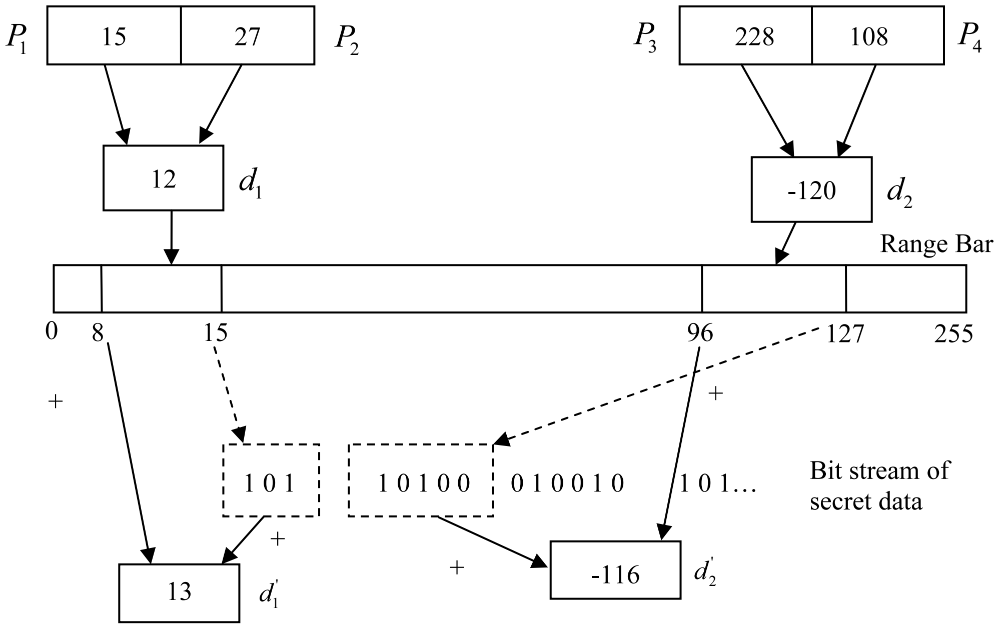


Fig. 3. An illustration of the data embedding process

data. The first secret data will be hidden is S_5^1, and second is S_5^2 and so on; and so forth. Next, we take the average number a_1 and a_2 to calculate the extraction function f_j as a weighted sum modulo 5 by Eq. (9).

$$f_j = (1 \times a_{2j-1} + 2 \times a_{2j}) \bmod 5 \tag{9}$$

No modification is needed if a secret digit S_5^j equals the extraction function value f_j of the block. When $f_j \neq S_5^j$, the modification rule is followed by using Zhang and Wang's scheme according to Section 2.1. A simple example of modifying the average number for hiding secret data is shown in Table 3. The Table 3 demonstrates that our scheme is much simple and has better performance in reducing cover image distortions, because we only need to adjust neither a_1 or a_2 by increase or decrease 1.

Table 3. An illustration of the proposed algorithm modifying the a_1 and a_2 of a block

Secret data S_5^1 (5-ary notation)	$a_1 = 15$	$a_2 = 27$	$f_1(a_1,a_2) = 4$
0	+1	No modify	0
1	No modify	+1	1
2	No modify	-1	2
3	-1	No modify	3
4	No modify	No modify	4

3.3 Embedding Procedure—Obtain New Pixel Value

After embedding all secret data into cover image, we need to make up the stego-image. Let us using d_1', d_2' and a_1', a_2' to obtain new pixel values P_1', P_2', P_3', P_4', according to the following procedure.

Case 1: $d_i^' \geq 0$

$$\Rightarrow \begin{cases} P_{2i-1}^' = a_i^' - \left\lfloor \dfrac{d_i^'}{2} \right\rfloor \\ P_{2i}^' = a_i^' + \left\lceil \dfrac{d_i^'}{2} \right\rceil \end{cases} \text{for } i = 1,2$$

Case 2: $d_i^' < 0$

$$\Rightarrow \begin{cases} P_{2i-1}^' = a_i^' + \left\lceil \dfrac{|d_i^'|}{2} \right\rceil \\ P_{2i}^' = a_i^' - \left\lfloor \dfrac{|d_i^'|}{2} \right\rfloor \end{cases} \text{for } i = 1,2$$

For example, the original pixel values $P_1 = 15, P_2 = 27, P_3 = 228, P_4 = 108$ of a block, after embedding secret data, assume that $d_1^' = 13, a_1^' = 21; d_2^' = -116, a_2^' = 168$. Now we can follow the above procedure to find out $P_1^', P_2^', P_3^', P_4^'$. So $P_1^' = 21 - \left\lfloor \dfrac{13}{2} \right\rfloor = 15$, $P_2^' = 21 + \left\lceil \dfrac{13}{2} \right\rceil = 28$ and $P_3^' = 168 + \left\lceil \dfrac{|-116|}{2} \right\rceil = 226$, $P_4^' = 168 - \left\lfloor \dfrac{|-116|}{2} \right\rfloor = 110$.

3.4 Embedding Procedure—Overflow Adjustment

However, in some cases, $P_1^', P_2^', P_3^', P_4^'$ may exceed the range [0,255]. For example, $P_1 = 255, P_2 = 255,$ $P_3 = 255,\ P_4 = 255$, in order to embed the secret data $S_5^1 = 1$, we need to increase a_1 by 1, so that the $f_1(a_1, a_2) = f_1(256, 255) = S_5^1 = 1$, but this modification may cause $P_1^'$ and $P_2^'$ overflow. In order to deal with this out of range problem, we need to re-adjust the pixel values to a proper value to satisfy the embedded secret data.

When we embed secret data by PVD, we will employ the Eq. (5) to detect such falling-off-boundary cases, and abandon the blocks which yield such cases for data embedding. In EMD step, in order to avoid the tricky block like $P_1 = 0, P_2 = 255, P_3 = 255, P_4 = 0$. When $d_1^' > 196\ or\ d_2^' > 196$, then do not embed secret data into this block, the block will left intact. It is note that such special blocks are very few in real applications. On the contrary, when any one of $P_1^', P_2^', P_3^', P_4^'$ exceeding the range [0,255], after embedding secret data, required re-adjust to satisfy the proper conditions. Next, we offer an example to show our mechanism of keeping the pixel values from exceeding the range [0,255] after secret data embedding. As shown in Table 4. Assume that, $a_1 = 255, a_2 = 0$, if secret data $S_5^1 = 1$ or $S_5^1 = 3$, following by EMD embedding algorithm, we should increase a_1 by 1 or decrease a_2 by 1, but this modification will make a_1 and a_2 overflow. For the purpose of both embedding secret data successfully and overcome overflow problem, we can re-adjust a_1 by decrease 4 and re-adjust a_2 by increase 4. After that the values of a_1, a_2 and $P_1^', P_2^', P_3^', P_4^'$ will fall within the range of [0,255].

Table 4. An illustration of the proposed algorithm overflows adjustment

Secret data (5-ary notation)	$a_1 = 255$	$a_2 = 0$	Overflow	Re-adjust a_1'	Re-adjust a_2'
0	No modify	No modify	No	No modify	No modify
1	+1	No modify	Yes	-4	No modify
2	No modify	+1	No	No modify	No modify
3	No modify	-1	Yes	No modify	+4
4	-1	No modify	No	No modify	No modify

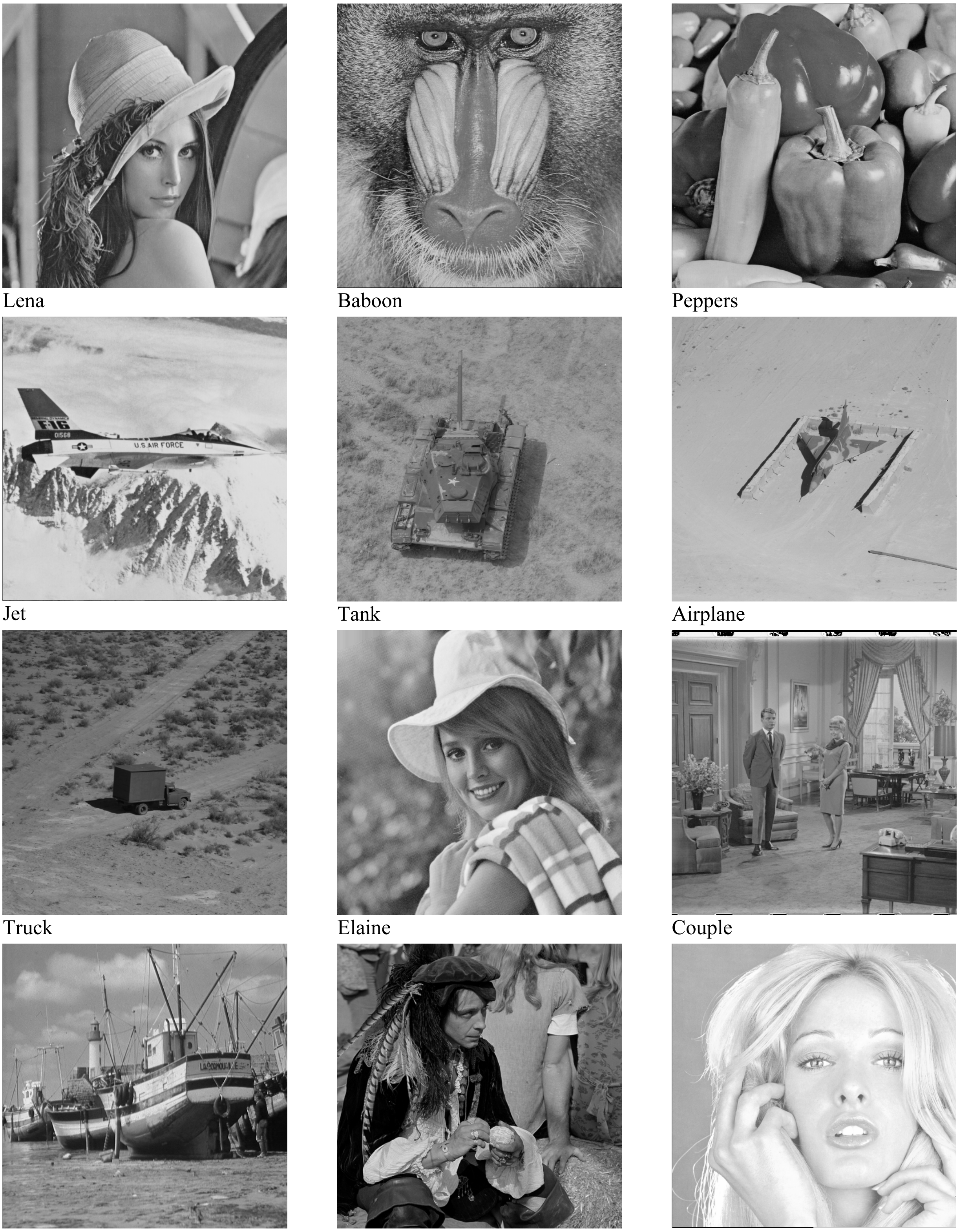
3.5 Extracting Procedure

In the recovery process, we can easily extract the secret data without using the original image. We first partition the stego-image into a series of non-overlapping blocks which contains four consecutive pixels, and select each sub-image by using the same direction as the embedding procedure one by one. However, when extracting secret data which embedded by PVD, it is essential to use the original range table R_j designed in the embedding phase in order to figure out the embedding capacity for each block B_k. Given a stego-block B_k with two two-pixel pairs as in the embedding process. Each time we visit these two pairs of a block in the stego-image. We apply the Eq. (5) to find out whether these two pairs were used or not in the embedding process. Then the extracting procedure is same as mentioned previously of Wu and Tsai's scheme on Section 2.1. For example, a stego-block B_k with four consecutive pixels from the stego-image with their pixel values being P_1', P_2', P_3' *and* P_4' respectively, assume that $P_1' = 15, P_2' = 28$, P_3'=226 *and* $P_4' = 110$. Then we can obtain $d_1' = 13$ *and* $d_2' = -116$ by Eq. (1). And the hidden capacity is 3 bits and 5bits, individually. Hence, the secret data's decimal values can extract by Eq. (6). Then convert the decimal values b_i' *and* b_{i+1}' gained by Eq. (6) into a binary string. Finally, we have secret data $5_{(10)} = 101_{(2)}$ and $21_{(10)} = 10100_{(2)}$, separately.

Extracting the secret data embedded by EMD is quite simple and easy. Given a stego-block B_k *and* , we can gained the d_1' *and* d_2', then in order to know whether the stego-block B_k was used or not in the embedding process of EMD step, when $d_1' > 196$ *or* $d_2' > 196$, it means this block was not used for embedding secret data. On the contrary, if it was used in the embedding process, then we can extract the secret data. First, obtain the average number a_1' *and* a_2', then we can find out the secret data S_5^j by Eq. (9). For example, assume a_1'=15 *and* $a_2' = 27$ of a stego-block, hence, we have secret data $f_1 = S_5^1 = 4$ by Eq. (9).

4 Experimental Results and Analysis

In this section, we shall present our experimental results to demonstrate the proposed method can perform better than Wu and Tsai's scheme. In our experiments, twelve gray-level cover images "Lena", "Baboon", "Peppers", "Jet", "Tank", "Airplane", "Truck", "Elaine", "Couple", "Boat", "Man", "Tiffany", shown in Fig. 4. were used as test images in our experiments. We used a series of pseudo random numbers as the secret data to be embedded into the cover images. We designed five sets of widths of ranges of gray value differences were used in the experiments. The peak signal-to-noise ratio (PSNR) was utilized to evaluate the stego-image quality, and the bit per pixel (bpp.) is used to evaluate the payload capacity. Generally speaking a larger PSNR values means a small difference between the stego-image and the original cover image. However the difference is unnoticeable by the human eye when the PSNR value is higher than 30 dB. In our experiments, the average PSNR values for all test images processed by the proposed scheme turn out to be higher than 30 dB with five sets of widths of ranges of gray value differences, which means the distortion caused by the processing of the proposed scheme was imperceptible to the human eye. The experiment results are given in Fig. 5. and Fig. 6.



Lena Baboon Peppers

Jet Tank Airplane

Truck Elaine Couple

Boat Man Tiffany

Fig. 4. Twelve gray-level cover image

The overall payload of each image was different according to the complexity of the cover image. The higher the complexity of the cover image, the higher the payload. The image “Baboon” is an example whose cover image complexity was high and payload was much higher than others. The rich edge areas in the “Baboon” image led to a relatively big number of pixel-value difference. In contrast to high complexity images, smoother images like “Tiffany” tend to have less payload of each block. Of course, on top of it, the sets of widths of ranges of gray value difference play a dominating role in deciding the pay

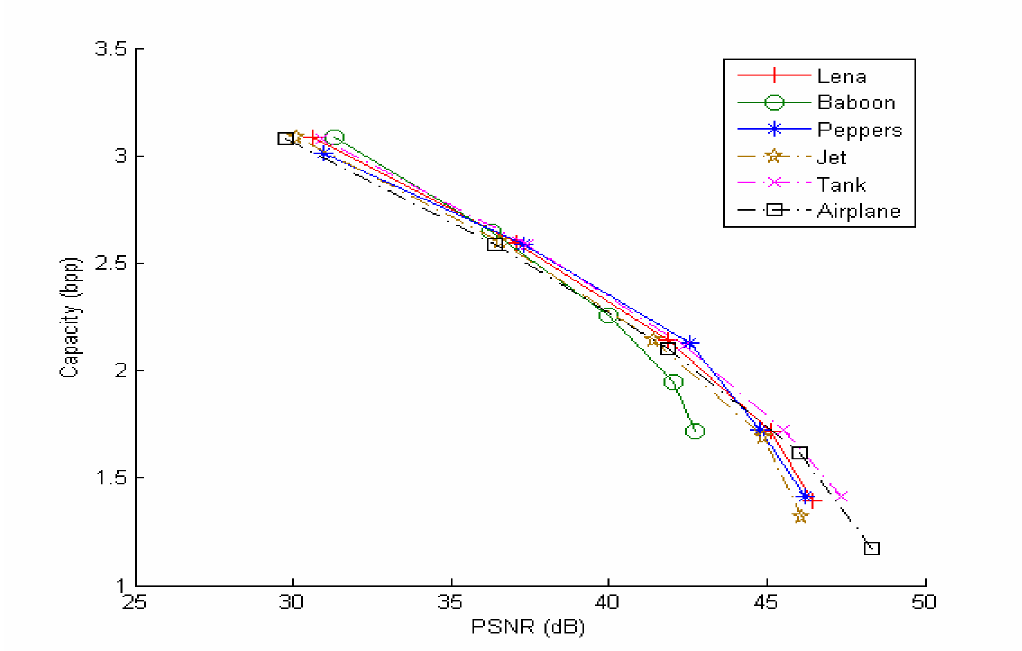


Fig. 5. The hiding capacity versus PSNR value for six tested images by the proposed scheme with five sets of widths of ranges of gray value differences

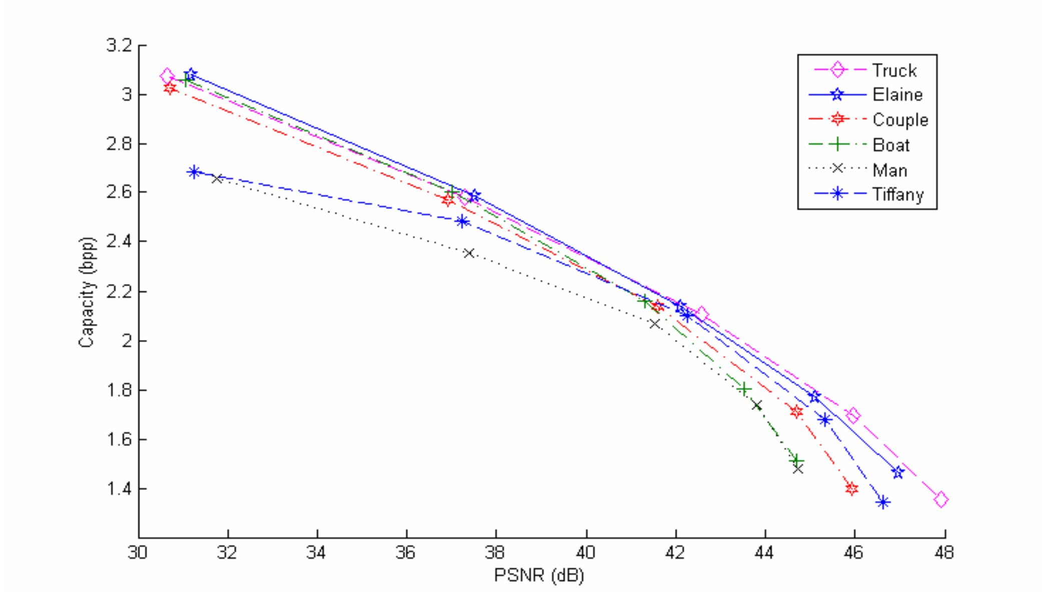


Fig. 6. The hiding capacity versus PSNR value for another tested images by the proposed scheme with five sets of widths of ranges of gray value differences

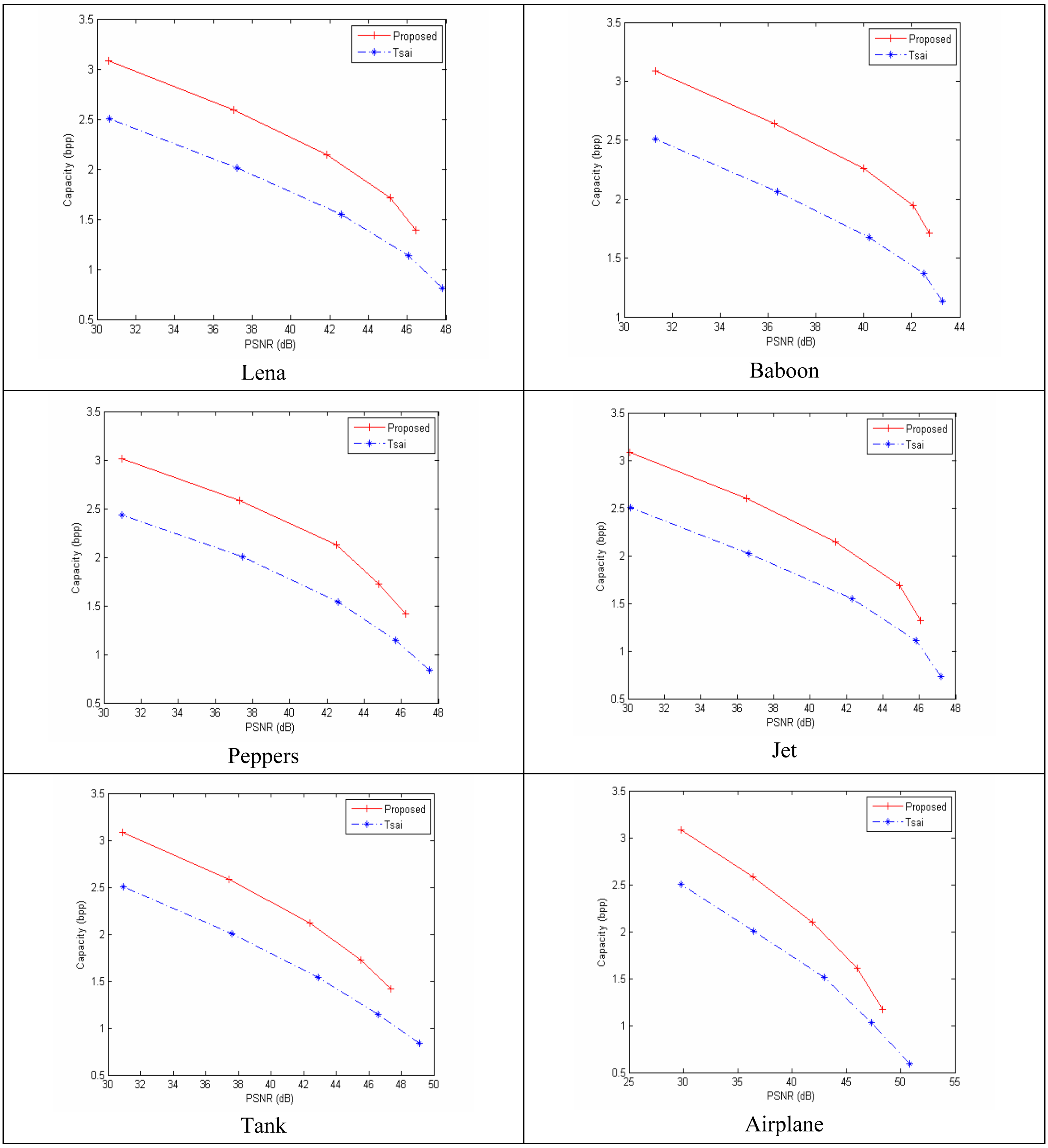
load and the stego-image quality for each image.

Furthermore, we have also compared the proposed scheme with the pixel value difference scheme by Wu and Tsai [20], we use the twelve gray-level cover images as tested image with five different width, in each values of PSNR, payload and bpp. are accounted by average, the results are illustrated in Table 5. In Table 5, we see that although the Tsai's scheme values of PSNR is a little higher than us, but our proposed scheme almost offer about 0.6 bpp additionally, on the other hand, we can say that based on the same quality of stego-image, our proposed scheme can provide more payload capacity than Tsai's scheme. This is because we reuse the average number of two pixel values of each block.

Moreover, we compare the proposed scheme with Tsai's scheme in terms of the maximal payload according to various PSNR values. The comparison results are shown in Fig. 7. and Fig. 8. In Fig. 7 and Fig. 8, we prove that our proposed scheme is superior to Wu and Tsai's scheme. Finally, in Fig. 9, we show

Table 5. The result of embedding the same random message in each five different width by Wu and Tsai's scheme including the proposed scheme

Range width	Wu and Tsai's scheme [20]			Our scheme		
	PSNR (dB)	Payload (bits)	bpp	PSNR (dB)	Payload (bits)	bpp
2, 2, 4, 4, 4, 8, 8, 16, 16, 32, 32, 64, and 64	47.5735	218,610	0.8339	46.1286	370,760	1.4143
4, 4, 8, 8, 8, 16, 16, 32, 32, 64, and 64	45.6637	302,510	1.1541	44.7395	454,690	1.7345
8, 8, 16, 16, 16, 32, 32, 64, and 64	42.3168	371,480	1.5453	41.6302	561,550	2.1422
16, 16, 32, 32, 32, 64, and 64	37.1679	520,120	1.9841	37.0265	672,270	2.5608
32, 32, 64, 64, and 64	30.8855	634,460	2.4202	30.8573	786,600	3.0007



Lena

Baboon

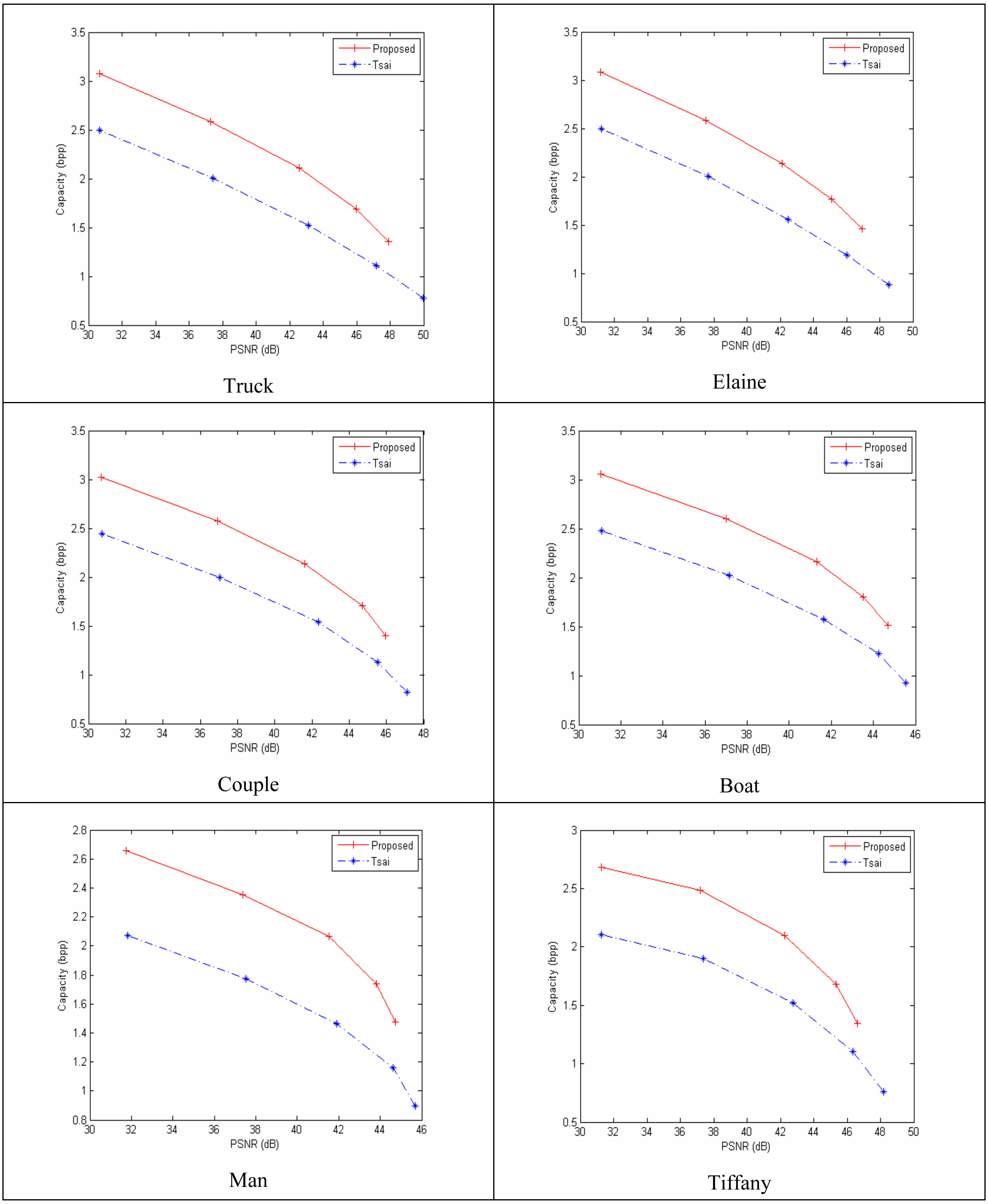
Peppers

Jet

Tank

Airplane

Fig. 7. Comparison results of Wu and Tsai's scheme and our scheme for six test images



Truck

Elaine

Couple

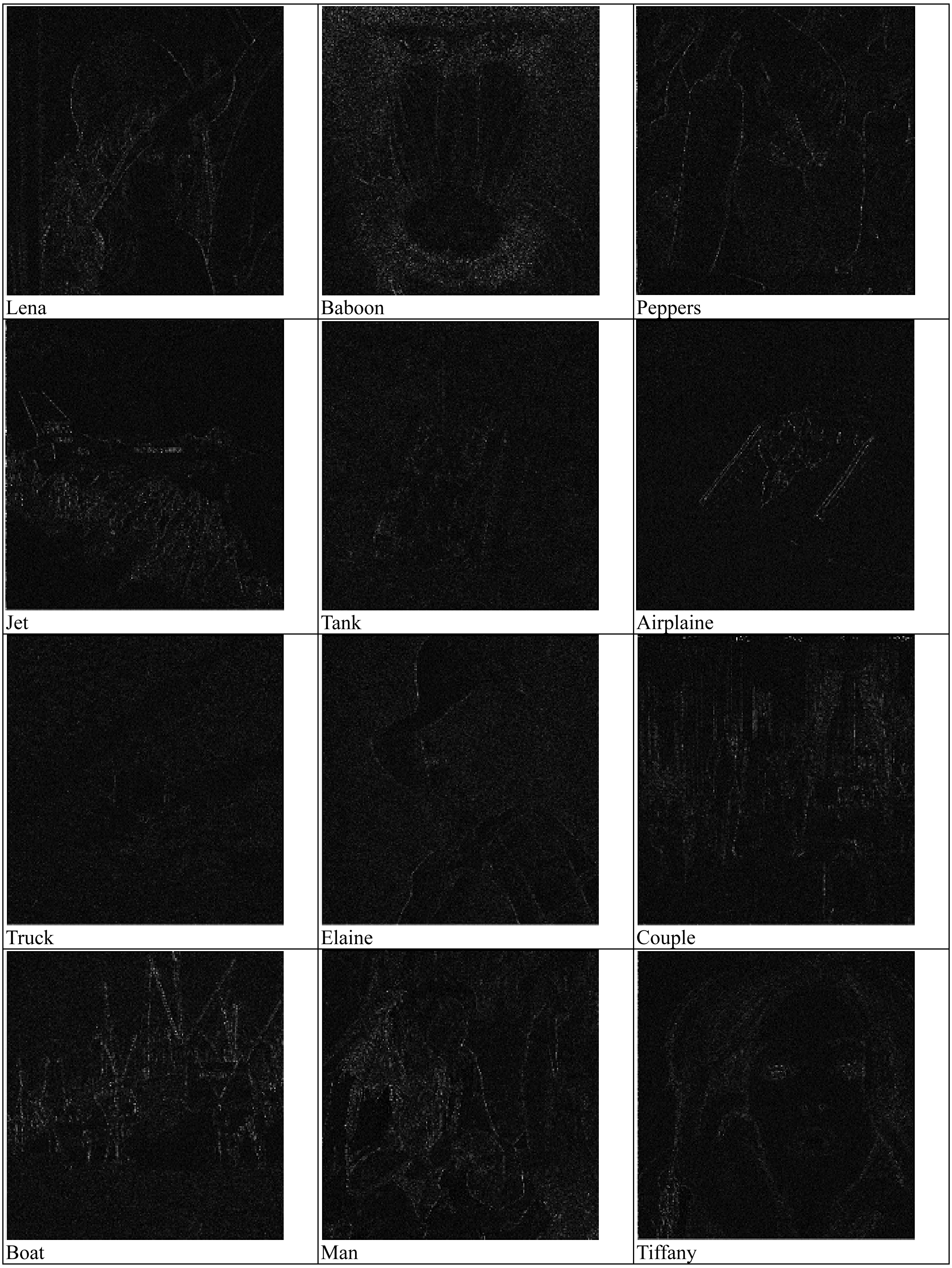
Boat

Man

Tiffany

Fig. 8. Comparison results of Wu and Tsai's scheme and our scheme for another test images

the corresponding enhanced difference images between the stego-image and the cover image, with the differences of gray values being scaled fifteen times. The enhanced difference images are shown here to indicate the distortions resulting from the embedding process. From them, we see that most of the distortions are found on the edges in the images. This means that such distortions will be less noticeable because changes in edge parts of images are generally less obvious to human eye.



Lena
Baboon
Peppers
Jet
Tank
Airplaine
Truck
Elaine
Couple
Boat
Man
Tiffany

Fig. 9. The enhanced difference images between the stego-image and the cover image of twelve test images

Finally, we compare the proposed scheme with other hiding schemes in terms of the payloads and various PSNR values. The comparison results are shown in Table 6. From the comparison results, we also prove that our proposed scheme has better performance regarding to the payloads and PSNR values.

Table 6. The comparison results of the proposed scheme with three inforation hiding schemes

	Payload (bits)	PSNR (dB)
Li and Wang's scheme [24]	73,728	37.06
Yang's scheme [25]	131,072	34.97
Jung and Yoo's scheme [26]	397,578	41.21
Our scheme	561,550	41.63

5 Conclusions

In this paper, we have proposed a novel scheme capable of providing a high payload capacity while maintain stego-image quality. The proposed scheme utilizes the remainder of the average value of the two consecutive pixels, which after PVD embedding process to embedding more secret data. The scheme not only provides a better way for embedding large amounts of data into cover images with imperceptions, but also uses the characteristic of the human vision's sensitivity to gray value variations. Besides, the proposed scheme also takes the falling-off-boundary problem into consideration, and the proposed scheme can solve the problem by re-revises the remainder of the average value of the four consecutive pixels. Experimental results show the proposed scheme has a much better performance than other schemes both in terms of cover image payload capacity and stego-image quality. The limitation of the proposed scheme is for embedding secret messages into a gray-valued cover image. In the future, we will also apply new technologys into information hiding in color image or design an efficient scheme regarding to the high payloads and high PSNR values.

Acknowledgements

The authors would like to express their appreciation to the anonymous referees for their valuable suggestions and comments. This research was partially supported by the Ministry of Science and Technology, Taiwan, R.O.C., under contract no.: MOST 104-2221-E-030-002.

References

[1] D. Artz, Digital steganography: hiding data within data, IEEE Internet Computing 5(3)(2001) 75-80.

[2] J. Zhao, E. Koch, C. Luo, Digital watermarking in business today and tomorrow, Communication of ACM 41(7)(1998) 67-74.

[3] A. Anees, A.M. Siddiqui, J. Ahmed, I. Hussain, A technique for digital steganography using chaotic maps, Nonlinear Dynamics 75(4)(2013) 807-816.

[4] S.M.R. Farschi, H. Farschi, A novel chaotic approach for information hiding in image, Nonlinear Dynamics 69(4)(2012) 1525-1539.

[5] M. Aziz, M.H. Tayarani-N, M. Afsar, A cycling chaos-based cryptic-free algorithm for image steganography, Nonlinear Dynamics 80(3)(2015) 1271-1290.

[6] M.U. Celik, G. Sharma, E. Saber, A.M. Tekalp, Lossless watermarking for image authentication: a new framework and an implementation, IEEE Transactions on Image Processing 15(4)(2006) 1042-1049.

[7] S. Dumitrescu, X. Wu, A new framework of LSB steganalysis of digital media, IEEE Transactions on Signal Processing 53(10)(2005) 3936-3947.

[8] L. Kamstra, H.J.A.M. Heijmans, Reversible data embedding into images using wavelet techniques and sorting, IEEE Transactions on Image Processing 14(12)(2005) 2082-2090.

[9] L.M. Marvel, C.G. Boncelet, C.T. Retter, Spread spectrum image steganography, IEEE Transactions on Image Processing 8(1999) 1075-1083.

[10] Y.C. Tseng, Y.Y. Chen, H. K. Pan, A secure data hiding scheme for binary images, IEEE Transactions on Communications 50(8)(2002) 1227-1231.

[11] X. Zhang, S. Wang, Steganography using multiple-base notational system and human vision sensitivity, IEEE Signal Processing Letters 12(1)(2005) 67-70.

[12] Z. Liao, Y. Huang, C. Li, Research on data hiding capacity, International Journal of Network Security 5(2)(2007) 140-144.

[13] G.J. Simmons, The prisoners' problem and the subliminal channel, in: Proce. of Crypto'83, 1984.

[14] N.I. Wu, M.S. Hwang, Data hiding: current status and key issues, International Journal of Network Security 4(1)(2007) 1-9.

[15] C.K. Chan, L.M. Cheng, Improved hiding data in images by optimal moderately significant-bit replacement, IEE Electronics Letters 37(16)(2001) 1017-1018.

[16] C.K. Chan, L.M. Cheng, Hiding data in images by simple LSB substitution, Pattern Recognition 37(3)(2004) 469-474.

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

[18] R.Z. Wang, C.F. Lin, J.C. Lin, Hiding data in image by optimal moderately significant-bit replacement, IEE Electronics Letters 36(25)(2000) 2069-2070.

[19] 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.

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

[21] X. Zhang, S. Wang, Efficient steganographic embedding by exploiting modification direction, IEEE Communications Letters 10(11)(2006) 781-783.

[22] H.W. Tseng, H.S. Leng, A steganographic method based on pixel-value differencing and the perfect square number, Journal of Applied Mathematics 2013.

[23] S.R. Tsui, C.T. Huang, W.J. Wang, A new adaptive steganographic method based on gradient adjacent prediction and side-match vector quantization, Journal of Information Hiding and Multimedia Signal Processing 4(4)(2014) 215-224.

[24] X. Li, J. Wang, A steganographic method based upon JPEG and particle swarm optimization algorithm, Information Sciences 177(2007) 3099-3109.

[25] C.H. Yang, Inverted pattern approach to improve image quality of information hiding by LSB substitution, Pattern Recognition 41(2008) 2674-2683.

[26] K.H. Jung, K.Y. Yoo, Data hiding method using image interpolation, Computer Standards & Inter faces 31(2009) 465-470.