Finding An Optimal Disturbed Square Matrix Using Dynamic Programming Strategy for Steganography

Chi-Shiang Chan, Yuan-Yu Tsai

[1] Department of M-Commerce and Multimedia Applications, Asia University, Taiwan
[2] Department of Medical Research, China Medical University Hospital, China Medical University, Taiwan
{CSChan, yytsai}@asia.edu.tw

Abstract

The purpose of this paper finds an optimal disturbed square matrix for steganography. The method is inspired from square matrix encoding (SME). SME obtains stego images by embedding secret data into cover images through the square matrix. To improve security and image quality, dynamic programming strategy is applied to disturb the digits in the square matrix on the basis of a given cover image and secret data. Through the disturbed square matrix, the stego images with the best image quality can be obtained. Moreover, the embedded data can not be extracted without the corresponding disturbed square matrix. Therefore, the security of the embedding mechanism can be improved. The experimental results show that the proposed method is superior to the other related methods.

Keywords: Information hiding, Diamond encoding, Square matrix encoding, Dynamic programming strategy

1 Introduction

Digital steganography [1] is a technique that embeds data into meaningful multimedia data. The embedded data can be secret data for the purpose of data transfer. Moreover, the embedded data can also be related data of images for the purpose of image management. The scenario of image management by digital steganography may occur in Blockchain. The property of Blockchain is that credentials (in the form of images) recorded on the Blockchain cannot be altered, faked or spoofed. Therefore, the information related to credentials reveled in Blockchain is valid. However, some information of credentials may be confidential. It only needs to record on Blockchain and to be extracted by some special persons or companies. In this case, digital steganography can be involved. The confidential information is embedded in credentials. Then, the credentials are recorded on the Blockchain to achieve the goal.

The simplest way to do digital steganography replaces the least-significant bits (LSBs) with data [2], which is called LSB Replacement. Although LSB Replacement has high hiding capacity, it usually causes image degradation. To overcome the drawback, some methods were proposed [3-5] to improve image quality. After that, Mielikainen's LSB matching revisited (LSBMR) [6] was proposed in 2006 to embed two bits into a pixel pair by subtracting/adding one from/to one of two pixels. Through inspecting the mechanism of LSBMR, it can be seen that the only one of two pixels is modified, and two secret bits can be embedded. Compared with other methods [3-5] of eliminating image degradation, LSBMR is superior because of this property. To further reduce image distortion, Chan's method [7] modified Mielikainen's method to link the bits with an exclusive-or (XOR) operator. Owing to linking operator, Chan's method only needs to modify some bits, and large amount of secret bits can be embedded. Chan's method can not only retain the embedding capacity as LSBMR but also reduce image distortion.

On the contrary, some methods focused on increasing the embedding capacity. In 2006, Zhang and Wang's method [8] proposed an exploiting modification direction (EMD) to embed (2n + 1)-ary secret data into cover pixels. Then, Chao et al.'s diamond encoding (DE) [9] refined EMD method to increase the quantity of the embedded secret data. Chao et al.'s diamond encoding modified pixels with the distance value between the diamond characteristic value (DCV) and the corresponding secret value. Through the distance value and the diamond matrix, the way of modifying cover pixels could be calculated. After that, Chen et al. [10] transformed the diamond matrix to a square matrix so as to reduce distortion which is caused by diamond encoding.

From the mechanism of diamond encoding and square matrix encoding, it can be seen the matrices used in these two encoding methods are fixed. But this may cause the security problem. The reason is that fixed matrices can be derived by everyone according to the same procedures. Once stego images are obtained,

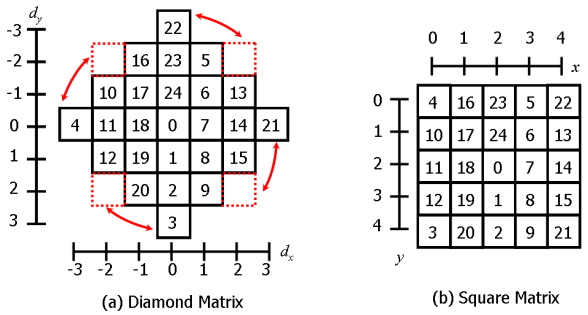
*Corresponding Author: Yuan-Yu Tsai; E-mail: yytsai@asia.edu.tw
DOI: 10.3966/160792642018111906029

the secret data can be extracted from them through the derived fixed matrices. Therefore, fixed matrices may cause the security problem. In order to improve the security and reduce the image distortion, Chan et al. [11] proposed a concept of dynamic matrix to replace the fixed matrices. For a given cover image and the secret data, a heuristic method is applied on the square error matrix to obtain a good disturbed square matrix. Through the good disturbed square matrix, image distortion could be further reduced. Because Chan et al.'s method is a heuristic method, the produced disturbed square matrix can not be guaranteed to be the best one. In this paper, the proposed method applied dynamic programming strategy on the square error matrix to obtain a best disturbed square matrix. Through the obtained disturbed square matrix, the image quality can be guaranteed as the best one among all possible stego images produced from other square matrices.

The rest of this paper is organized as follows. Diamond encoding and square matrix encoding are described in Section 2. The details of the proposed method are presented in Section 3. Section 4 demonstrates the experimental results. Finally, some conclusions are drawn in Section 5.

2 Related Works

This section introduces Chen et al.'s square matrix encoding [10]. In 2013, Chen et al. modified diamond encoding [9] and developed their square matrix encoding to reduce the image distortion. DE and SME embedded the secret data by referring to a diamond matrix and a square matrix, respectively. Both two matrices are shown in Figure 1. Because outermost points (3, 0), (-3, 0), (0, 3), and (0, -3) are far away from the center point of coordinates in the diamond matrix, DE may cause great distortion. In Chen et al.'s method, the diamond matrix is modified as the square matrix, and the corresponding hiding method is also modified to fit the square matrix.



(a) Diamond Matrix (b) Square Martix

Figure 1. The diamond matrix and square matrix

The embedding procedures of square matrix encoding by using the square matrix are introduced in the following paragraphs. First of all, the secret binary data is converted to a numerical base with base 25 for the given secret data, and the i-th secret digit is denoted as S_i. As for the cover image, two non-overlapping neighboring pixels p_{2i} and p_{2i+1} are taken as a pixel pair, and the i-th pixel pair is denoted as P_i. The i-th secret digit S_i will be embedded into the i-th pixel pair P_i. By referring the square matrix in Figure 1(b), the coordinate (x_i, y_i) of a cover pixel pair in the square matrix is (p_{2i} mod 5, p_{2i+1} mod 5). Meanwhile, the coordinate (x_i', y_i') that makes $SM(x_i', y_i')$ equal to S_i can also be known. The distance between two coordinates can be calculated as:

$$\begin{cases} d_i^x = x_i' - x_i \\ d_i^y = y_i' - y_i \end{cases} \quad \textbf{(1)}$$

Normally, two stego pixels can be obtained by adding d_i^x and d_i^y to p_{2i} and p_{2i+1}, respectively. However, in order to reduce the distortion, two different values d_i^x and d_i^y can be further modified as:

$$\begin{cases} d_i^x = d_i^x + 5, & if \quad d_i^x < -2, \\ d_i^x = d_i^x - 5, & if \quad d_i^x > 2, \\ d_i^x = d_i^x, & otherwise. \end{cases} \quad \textbf{(2)}$$

$$\begin{cases} d_i^y = d_i^y + 5, & if \quad d_i^y < -2, \\ d_i^y = d_i^y - 5, & if \quad d_i^y > 2, \\ d_i^y = d_i^y, & otherwise. \end{cases} \quad \textbf{(3)}$$

Finally, a stego pixel pair p'_{2i} and p'_{2i+1} can be obtained by the following equation:

$$\begin{cases} p'_{2i} = p_{2i} + d_i^x \\ p'_{2i+1} = p_{2i+1} + d_i^y \end{cases} \quad \textbf{(4)}$$

In the extracting phase, the i-th secret digit S_i can be extracted by referring to the values in $SM(p'_{2i} \bmod 5, p'_{2i+1} \bmod 5)$.

Through the above description, it shows that the square matrix is needed in the encoding and decoding sides. Since the square matrix is derived from the diamond matrix, the square matrix can be produced easily. If an attacker knows there has something in the stego image, he/she will extract the secret data from this stego image. Encrypting the secret data before embedding may be a good way to solve this problem. However, an additional burden will be needed. In 2014, Chan et al. [11] proposed a method to disturb the digits in the square matrix so that the secret data can not be extracted by attackers. The disturbed square matrix is treated as a secret key and transferred to the receiver through a secure channel. Furthermore, the way of disturbing the digits may affect the quality of the stego images. However, Chan et al.'s [11] method can only produce good disturbed square matrix. In the next

section, the proposed method applied dynamic programming strategy to obtain an optimal disturbed square matrix. Through the optimal disturbed square matrix, the image quality of the stego images in the proposed method is better than that in the other methods.

3 Proposed Method

There are two main stages in the proposed method. The first stage produces an optimal disturbed square matrix according to a given cover image and the given secret data. The second stage uses the same procedures as Chen et al.'s square matrix encoding [10] to embed the secret data into the cover image. Since the second stage is the same as Chen et al.'s method, the embedding procedure will not be described here redundantly. We focus on the way to get an optimal disturbed square matrix such that the best quality of the stego image can be obtained by replacing the square matrix in Chen et al.'s method with the optimal disturbed square matrix. There are two steps. The first step produces a square error matrix according to a given cover image and the given secret data. The second step produces an optimal disturbed square matrix through a square error matrix using dynamic programming strategy. The details of these two steps are described in the following two subsections.

3.1 Produce a Square Error Matrix

In this subsection, the square error matrix is produced using the method in [11]. Note that Chan et al.'s method [11] derived a disturbed square matrix produced from square error matrix using a heuristic algorithm. Therefore, Chan et al.'s method only got an approximately optimal disturbed square matrix. In order to get an optimal disturbed square matrix, the proposed method uses dynamic programming strategy in the square error matrix. That is the reason why we introduce the way of producing a square error matrix according to a given cover image and the given secret data in the first step.

First of all, a disturbed square matrix is defined as $D_{N\times N}=\{d[i][j]|\ 0\leqq i, j<N\}$ and a square error matrix is defined as $M_{N^2\times N^2}=\{m[i][j]|\ 0\leqq i, j< N^2\}$. The initial values of all elements in the square error matrix are equal to 0. The content of $m[i][j]$ represents the sum of square errors when the digit with value i is put at the position $(\lfloor j/N \rfloor, j \bmod N)$ in the disturbed square matrix. In the following paragraphs, N is set 5 as an example to demonstrate the whole procedures. Since N is equal to 5, the size of the disturbed square matrix is 5 by 5, and the size of the square error matrix is 25 by 25. The positions of all elements in $D_{5\times5}$ are marked from position 0 to 24 sequentially in the way of row major.

More precisely, the secret binary data is converted to a numerical base with base 25 for the given secret data, and the i-th secret digit is denoted as S_i. As for the cover image, two non-overlapping neighboring pixels p_{2i} and p_{2i+1} are taken as a pixel pair, and the i-th pixel pair is denoted as P_i. The i-th secret digit S_i will be embedded into the i-th pixel pair P_i. If the original coordinate (x_i, y_i) represents the cover pixel pair in the disturbed square matrix and its value is equal to (p_{2i} mod 5, p_{2i+1} mod 5), and the secret digit S_i is put at the coordinate $(\lfloor j/5 \rfloor, j \bmod 5)$ in the disturbed square matrix where j is in the range from 0 to 24, then the distance between two coordinates d_i^x and d_i^y can be obtained by Formula (1)-(3). Therefore, the equation to update $m[S_i][j]$ is given as:

$$m[S_i][j]=m[S_i][j]+(d_i^x)^2+(d_i^y)^2. \quad \textbf{(5)}$$

Since S_i can be put at the coordinates from (0, 0) to (4, 4), the value j can be from 0 to 24. That means for each secret digit S_i, all values in $m[S_i][j]$ should be updated where $0\leqq j<25$.

After updating the contents of the square error matrix according to all cover pixel pairs and their corresponding secret data, the final values of all elements in the square error matrix can be obtained.

For example, assume that the first two pixel pairs are (26, 25) and (25, 27), and both two secret digits are equal to 3. For the first pixel pair, the original coordinate (x_i, y_i) is equal to (1, 0) using (26 mod 5, 25 mod 5). Because the secret digits are equal to 3, the values from $m[3][0]$ to $m[3][24]$ should be updated. At the beginning, if we want to update $m[3][0]$, it means the digit with value 3 will be put in $(\lfloor j/5 \rfloor, j \bmod 5)$, that is, (0, 0). According to Formula (1)-(3), the distance between two coordinates d_0^x and d_0^y are 1 and 0, respectively. Therefore, $m[3][0]$ is updated as $m[3][0]+1^2+0^2$, that is, 1. Similarly, if we want to update $m[3][1]$, it means the digit with value 3 will be put in $(\lfloor j/5 \rfloor, j \bmod 5)$ in the disturbed square matrix, that is, (0, 1). Then, the distance between two coordinates d_0^x and d_0^y are 1 and 1, respectively. Therefore, $m[3][1]$ is update as 2. If the all values in $m[3][j]$ are updated where $0\leqq j<25$, the final result of the square error matrix is the same as Figure 2(a).

For the second pixel pair, the original coordinate (x_i, y_i) is equal to (0, 2) using (25 mod 5, 27 mod 5). Note that the second secret digit is also equal to 3. To update $m[3][0]$, distance values d_1^x and d_1^y can be obtained, and they are 0 and 2, respectively. Since the value of $m[3][0]$ is equal to 1 as shown in Figure 2(a), $m[3][0]$ is updated as $m[3][0]+0^2+2^2$, that is, 5. Similarly, distance values d_1^x and d_1^y are 0 and 1, when putting the digit with value 3 at (0, 1) in the disturbed square matrix. Therefore, $m[3][1]$ is updated as $m[3][1]+0^2+1^2$,

that is, 3. If the all values in $m[3][j]$ are updated where $0 \leqq j < 25$, the result of processing first two pixel pairs can be obtained as shown in Figure 2(b). If the procedures are performed on each pixel pair and its corresponding secret digit, the final square error matrix will be obtained.

	0	1	2	3	...	24
0	0	0	0	0	...	0
1	0	0	0	0	...	0
2	0	0	0	0	...	0
3	1	2	5	5	...	5
...	...	...	...	...	...	...
24	0	0	0	0	...	0

(a)

	0	1	2	3	...	24
0	0	0	0	0	...	0
1	0	0	0	0	...	0
2	0	0	0	0	...	0
3	5	3	5	6	...	10
...	...	...	...	...	...	...
24	0	0	0	0	...	0

(b)

Figure 2. The result of the square error matrix

3.2 Produce an Optimal Disturbed Square Matrix

Note that $m[S_i][j]$ means the sum of square errors when the digit with value S_i is put at the coordinate $(\lfloor j/N \rfloor, j \bmod N)$ in the disturbed square matrix. If we can select N^2 elements from the square error matrix under the condition that exactly one element is picked up in each row and column, and the sum of selected elements is smallest, then the square error between the cover image and the stego image is smallest.

More precisely, assume that the selected N^2 elements are $m[k][j_k]$ and $j_{k1} \neq j_{k2}$ if $k1 \neq k2$. If the value of $m[0][j_0]+ m[1][j_1]+\ldots+ m[N^2\text{-}1][\ j_{N^2-1}\ ]$ is the smallest, then putting 0, 1, …, N^2-1 in the positions $(\lfloor j_0/N \rfloor, j_0 \bmod N)$, $(\lfloor j_1/N \rfloor, j_1 \bmod N)$, …,$(\lfloor j_{N^2-1}/N \rfloor, j_{N^2-1} \bmod N)$ in the disturbed square matrix can cause the smallest sum of square errors. This also means that for each selected $m[k][j_k]$, the disturbed square matrix $D[\lfloor j_k/N \rfloor][j_k \bmod N]$ is equal to k. Therefore, if we can select N^2-1 elements to make the sum of their values the smallest under the previous described conditions, an optimal disturbed square matrix can be derived. By replacing the square matrix in Chen et al.'s method with an optimal disturbed square matrix, the best quality of the stego image can be obtained.

It is difficult to select the elements under the condition that exactly one element is picked up in each row and column such that the sum of the selected elements is the smallest. To achieve this goal, dynamic programming strategy can be applied. First of all, we set a variable *SET* which is used to represent subset of {0, 1, …, N^2-1}. The second variable *Min*[*r*, *SET*] is used to represent the smallest sum of the submatrix M' under the previous mentioned conditions where the submatrix M' is constructed by the rows of the square error matrix from 24 to *r* and the columns listed in *SET*. It can be seen *Min*[0, {0, 1,…, N^2-1}] represents the smallest sum of the whole square error matrix. Through the above definition, the recursive formula of dynamic programming strategy can be written as below:

$$Min[r, SET] = \min_{j \in SET} \{m[r][j] + Min[r, SET - \{j\}]\}. \quad \textbf{(6)}$$

Note that through this formula, the smallest value of each submatrix M' can be calculated and recorded. Once the smallest value is needed again, it can be extracted without re-calculating. Moreover, the smallest value of a large size submatrix comes from that of the small size submatrices. Therefore, it can ensure that the obtained value is smallest value. Since the smallest value is obtained, we can trace back to find out the elements that produce the smallest value. As the result, the disturbed square matrix can be obtained by these elements.

We provide a small example to demonstrate the above procedures. In this example, *N* is equal to 2. Then, the size of the disturbed square matrix is 2 by 2. In addition, the size of the square error matrix is 4 by 4 as shown in Figure 3. According to the previous definition, *Min*[0, {0, 1, 2, 3}]is the smallest values of the square error matrix. Its value comes from finding the smallest values among four cases as shown in Figure 3.

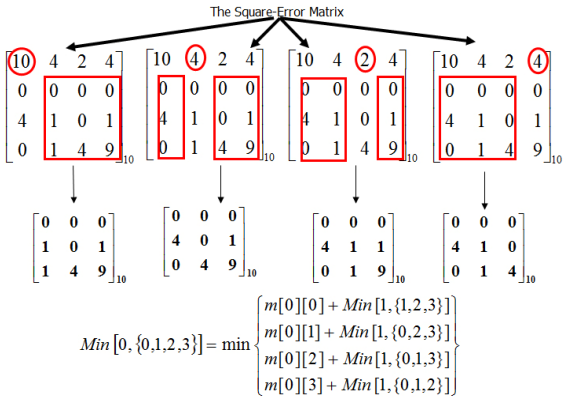


Figure 3. The procedures of the proposed method

Until now, the smallest values for *Min*[1,{1, 2,3}], *Min*[1,{0, 2,3}], *Min*[1,{0, 1, 3}], and *Min*[1,{0, 1, 2}] are still unknown. Therefore, the same procedures are performed recursively to calculate their value as shown in Figure 4.

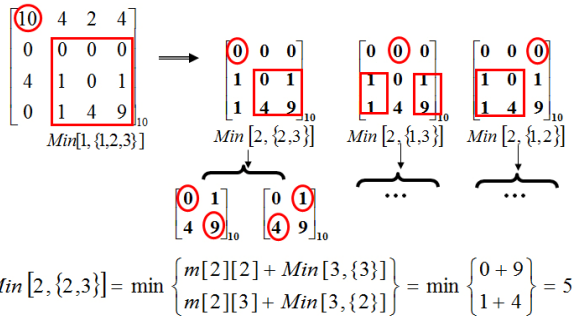


Figure 4. The procedures of the proposed method

According to the example in Figure 4, the smallest value for *Min*[2,{2,3}] can be obtained, that is, 5. The smallest value and the corresponding selected elements are recorded. Similarly, the smallest values and the corresponding selected elements for *Min*[2,{1,3}] and *Min*[2,{1,2}] are also recorded. Through the smallest values of *Min*[2,{2,3}], *Min*[2,{1,3}] and *Min*[2,{1,2}], the smallest values of *Min*[1,{1, 2,3}] can be calculated. Recursively, the smallest values of *Min*[0,{0, 1, 2, 3}] can be obtained. Moreover, because the smallest value of each submatrix is recorded, its smallest value won't need to be recalculated when it will be needed in the next time. For example, *Min*[2,{2,3}] is needed in two situations in Figure 5. Therefore, once the smallest value of each submatrix is recorded, it will be used directly in the next time.

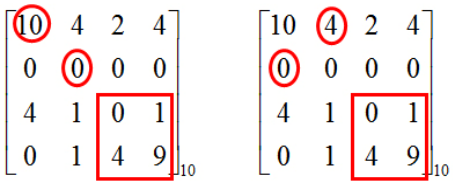


Figure 5. The procedures of the proposed method

Since the smallest values of the square error matrix can be obtained, the selected elements can be known by tracing back. According to the above example, the value of $m[0][2]+ m[1][3]+ m[2][1]+ m[3][0]$ is the smallest. Hence, the optimal disturbed square matrix ca be derived by putting value 0, 1, 2, and 3 in (1,0), (1,1), (0,1), and (0,0), respectively. The final result is shown in Figure 6. By replacing the square matrix in Chen et al.'s method with the optimal disturbed square matrix, the best quality of the stego image can be obtained.

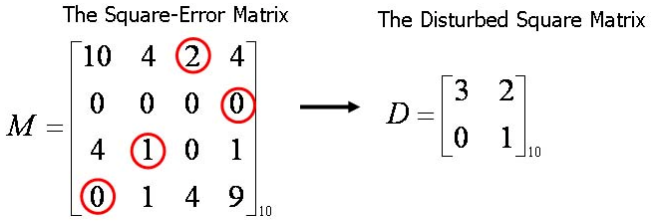
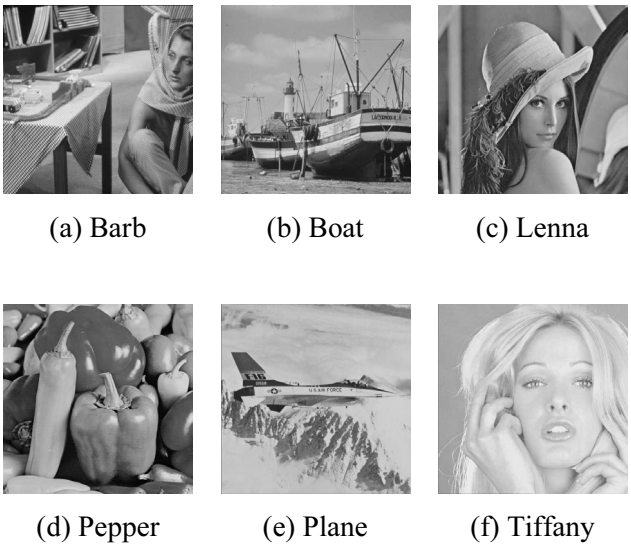


Figure 6. The final result

4 Experimental Results

The experimental results will be demonstrated in this section. In our experiment, the cover images were Barb, Boat, Lenna, Pepper, Plane and Tiffany with size 512×512 pixels, as shown in Figure 7. Normally, the secret binary data should be converted to a numerical base with base 25 for given secret data when N is equal to 5. However, for convenience, the secret images were derived from resizing all cover images to images with 256×512 pixels and pixel values in the resized cover pixel were adjusted proportionally from 0-255 to 0-24.



(a) Barb (b) Boat (c) Lenna

(d) Pepper (e) Plane (f) Tiffany

Figure 7. Six 512×512-pixel cover images

The way to estimate the quality of stego-images was the peak signal to noise ratio (*PSNR*), which can be calculated from the following formula:

$$PSNR = 10 \times \log \frac{(255)^2}{MSE} dB \quad \textbf{(7)}$$

Here *MSE* means the mean squared error, and it is derived from the square errors of all pixels.

$$MSE = \frac{1}{(w \times h)} \sum_{I=1}^{w} \sum_{J=1}^{h} (\alpha(I,J) - \beta(I,J))^2 \quad \textbf{(8)}$$

The symbols $\alpha(I, J)$ and $\beta(I, J)$ represent the pixel values at the position (I, J) in the stego-image and the original image, respectively. The symbols w and h represent the pixel numbers for the width and the height of the image, respectively.

The experimental results are shown in Table 1 to Table 3. The values in Table 1 are the average *MSE* values among the different methods when the secret images are embedded into cover images one by one.

Table 1. The average *MSE* among different methods

	Cover Images					
Method	Barb	Boat	Lenna	Pepper	Plane	Tiffany
DE [9]	545783	544954	545012	546561	547411	545801
SME [10]	524112	523317	524178	524055	523437	523278
ISME [11]	518232	517835	518331	518176	513307	514293
Proposed Method	518105	517713	518225	518004	512969	513610

For comparison, the distances between the proposed method and the other related methods are listed in Table 2. The values in Table 2 come from subtracting *MSE* values of the proposed method from those of the other related methods.

Table 2. The improvement comparing the proposed method with other related methods

	Cover Images					
Method	Barb	Boat	Lenna	Pepper	Plane	Tiffany
DE [9]	27679	27241	26787	28557	34443	32191
SME [10]	6007	5605	5953	6051	10469	9668
ISME [11]	127	122	106	172	338	683

Table 3. The *PSNR* values among different methods

	Cover Images					
Method	Barb	Boat	Lenna	Pepper	Plane	Tiffany
DE [9]	44.946	44.953	44.952	44.94	44.933	44.946
SME [10]	45.122	45.129	45.121	45.122	45.128	45.129
ISME [11]	45.171	45.174	45.17	45.171	45.212	45.204
Proposed Method	45.172	45.175	45.171	45.173	45.215	45.21

Through the experimental results as shown in Table 1 and Table 2, the *MSE* values of the stego images in the proposed method are always lower than those in the other methods. It goes without saying that the *PSNR* values of stego images in the proposed method are higher than those in the other methods, as shown in Table 3. This means the stego images produced by the proposed method have better image quality than that produced by other methods.

5 Conclusion

In order to improve security and reduce image distortion, Chan et al. [11] proposed a concept of a dynamic matrix to replace the fixed matrices. The dynamic matrix came from disturbing the digits in square matrix according to a given cover image and the secret data. However, Chan et al.'s method could not produce the optimal disturbed square matrix. In this paper, the proposed method used dynamitic programming strategy to get an optimal disturbed square matrix. Through the optimal disturbed square matrix, the best image quality of stego images can be obtained. According to experimental results, the *MSE* values of the stego images in the proposed method are always lower than those in the other methods.

Acknowledgments

This work was supported by Ministry of Science and Technology of Taiwan under the grant numbers MOST 105-2410-H-468-010, MOST 105-2221-E-468-019, and MOST 106-2632-E-468-003.

References

[1] L. C. Huang, L. Y. Tseng, M. S. Hwang, The Study of Data Hiding in Medical Images, *International Journal of Network Security*, Vol. 14, No. 6, pp. 301-309, November, 2012.

[2] C. K. Chan, L. M. Cheng, Hiding Data in Images by Simple LSB Substitution, *Pattern Recognition*, Vol. 37, No. 3, pp. 469-474, March, 2004.

[3] C. C. Chang, J. Y. Hsiao, C. S. Chan, Finding Optimal LSB Substitution in Image Hiding by Dynamic Programming Strategy, *Pattern Recognition*, Vol. 36, No. 7, pp. 1583-1595, July, 2003.

[4] C. C. Thien, J. C. Lin, A Simple and High-hiding Capacity Method for Hiding Digit-by-digit Data in Images Based on Modulus Function, *Pattern Recognition*, Vol. 36, No. 12, pp. 2875-2881, December, 2003.

[5] R. Z. Wang, C. F. Lin, J. C. Lin, Hiding by Optimal LSB Substitution and Genetic Algorithm, *Pattern Recognition*, Vol. 34, No. 3, pp. 671-683, March, 2001.

[6] J. Mielikainen, LSB Matching Revisited, *IEEE Signal Processing Letters*, Vol. 13, No. 5, pp. 285-287, May, 2006.

[7] C. S. Chan, On Using LSB Matching Function for Data Hiding in Pixels, *Fundamenta Informaticae*, Vol. 96, No. 1-2, pp. 49-59, January, 2009.

[8] X. P. Zhang, S. Z. Wang, Efficient Steganographic Embedding by Exploiting Modification Direction, *IEEE Communications Letters*, Vol. 10, No. 11, pp. 781-783, November, 2006.

[9] R. M. Chao, H. C. Wu, C. C. Lee, Y. P. Chu, A Novel Image Data Hiding Scheme with Diamond Encoding, *EURASIP Journal on Information Security*, Vol. 2009, Article ID 658047, pp. 1-9, December, 2009.

[10] J. Chen, C. W. Shiu, M. C. Wu, An Improvement of Diamond Encoding using Characteristic Value Positioning and Modulus Function, *The Journal of Systems and Software*, Vol. 86, No. 5, pp. 1377-1383, May, 2013.

[11] C. S. Chan, Y. H. Chen, Y. Y. Tsai, An Improvement of Square Matrix Encoding by Adjusting Digits in a Matrix, *International Journal of Network Security*, Vol. 16, No. 4, pp. 313-317, July, 2014.

Biographies

Chi-Shiang Chan received the Ph.D degree in computer engineering from National Chung Cheng University, Chiayi in 2005. He is currently a professor with the Department of M-Commerce and Multimedia Applications, Asia University. His research interests include image and signal processing, image compression, information hiding, and data engineering.

Yuan-Yu Tsai is currently an associate professor at the Department of M-Commerce and Multimedia Applications, Asia University, Taiwan. He is also a research consultant with the Department of Medical Research, China Medical University Hospital, China Medical University. His research interests include computer graphics and multimedia security.