An Improvement of Square Matrix Encoding by Adjusting Digits in a Matrix

Chi-Shiang Chan, Yi-Hui Chen, and Yuan-Yu Tsai
(*Corresponding author*: *Yuan-Yu Tsai*)

Department of Applied Informatics and Multimedia, Asia University
No. 500, Lioufeng Rd., Wufeng Dist., Taichung City 41354, Taiwan
(Email: yytsai @asia.edu.tw)

(Received Sep. 30, 2013; revised and accepted Oct. 6, 2013)

Abstract

The proposed method was derived from Chen's Square Matrix Encoding for information hiding. Square Matrix Encoding is modified from Diamond Encoding by changing a diamond matrix to a square matrix. The image distortion is reduced when the secret data were embedded by using the square matrix. In order to further improve the security and reduce image distortion, this paper proposes a mechanism to adjust the digits in the square matrix according to a given cover image and secret data. By employing the adjusted-square matrix, good quality stego images can be achieved. According to experimental results, the Square Error values between the stego images and the cover images in the proposed method are lower than those in the other two methods. The proposed method is thus superior to both Diamond Encoding and Square Matrix Encoding.

Keywords: Diamond encoding, information hiding, square matrix encoding

1 Introduction

Digital Hiding [6-8] is a technique that embeds secret data into meaningful multimedia data. It can be roughly classified into two types: frequency-based image hiding and pixel-based image hiding. Frequency-based image hiding embeds secret data into coefficients in the frequency domain, such as AC coefficients. Values changed in the high frequency domain are not easily detected by the human eye. On the contrary, pixel-based image hiding directly embeds secret data into pixels. One simple way of pixel-based image hiding is to replace the least-significant bits (LSBs) with secret data [2]. The benefit of pixel-based image hiding is that its hiding capacity is usually greater than frequency-based image hiding.

However, replacing LSBs with secret data may cause image degradation. In order to alleviate the degradation, Wang et al. [11] and Chang et al. [3] proposed their method of mapping the values of secret bits into other values that were closer to the values of the original cover bits. Moreover, a modulus function [10] was also used in pixels to embed secret data. Although the above methods could achieve high embedding capacity, the image quality of those methods was not as good as Mielikainen's LSB matching revisited (LSBMR) technique [9]. Mielikainen's method partitioned cover pixels into pixel pairs, and two secret bits were embedded in a pixel pair. For each pixel pair, only one pixel value was added or subtracted 1 to embed a 4-ary digit. Chan's method [1] used the same concept as Mielikainen's method to link the bits with an Exclusive OR (XOR) operator and used them to embed secret data. Chan's method could not only retain the embedding capacity but could also reduce image distortion. On the contrary, in order to increase the embedding capacity, Zhang and Wang's method [12] proposed an exploiting modification direction (EMD) to embed (2n + 1)-ary secret data into cover pixels. Following this, Chao et al.'s Diamond Encoding (DE) [4] method was derived from Zhang and Wang's method [12] to increase the payload without degrading image quality too much. The DE method calculated the diamond characteristic value (DCV) for a pixel pair. The distance value between DCV and the corresponding secret value could be obtained. Moreover, there was a diamond matrix in the DE method. Through distance value and the diamond matrix, the way to modify cover pixels could be ascertained. In 2013, Chen et al. [5] inspected the diamond matrix and indicated that some modifications inducted by the diamond matrix might cause a lot of distortion. Therefore, Chen et al. proposed their Square Matrix Encoding (SME) method in which the diamond matrix is modified into square matrix.

In order to improve security and reduce image distortion, this paper proposes a method to adjust the digits in the square matrix to form an adjusted-square matrix according to a given cover image and secret data. It is very important to obtain a good adjusted-square matrix because a good adjusted-square matrix can derive a good quality stego image. To achieve this goal, a square error matrix was produced. Then, the problem of finding a good

adjusted-square matrix is transformed into the problem of selecting elements in the square error matrix such that the sum of the selected elements was the smallest. Through the results of the selected elements in the square error matrix, a good adjusted-square matrix could be found. Through the good adjusted-square matrix, the image distortion could be further reduced.

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 The Related Work

This section introduces Chao et al.'s Diamond Encoding (DE) [4], and continues with a presentation of Chen et al.'s Square Matrix Encoding (SME) [5].

2.1 Diamond Encoding (DE)

Diamond Encoding first determines the parameter value k according to the quantity of secret data. The equation for determining parameter value k is shown below:

$$\left\lfloor \frac{H \times W}{2} \times \log_2\left(2k^2 + 2k + 1\right) \right\rfloor \geq |Secret| . \quad (1)$$

The symbols H and W denote the number of pixels at row and column in a cover image, respectively. Moreover, the symbol $|Secret|$ is used to represent the bit length of the secret data. Once the parameter value k is determined, the secret data is converted to a numeral base with B-ary where B is equal to $(2k^2+2k+1)$. As for the cover image, two non-overlapping neighboring pixels are taken as a pair to calculate the Diamond Characteristic Value (DCV). The equation to calculate the DCV is demonstrated below:

$$f(p_1, p_2) = ((2k+1) \times p_1 + p_2) \bmod B \quad (2)$$

where p_1 and p_2 represent two pixels in a pair. Now that the secret digit s_B in base B and the DCV of a cover pixel pair are obtained, the distance value d between two values can be calculated as:

$$d = (s_B - f(p_1, p_2)) \bmod B . \quad (3)$$

The position with value d in the diamond matrix can then be found. The diamond matrices under different kinds of k are shown in Figure 1. Take the center position as the origin of coordinates and use (d_x, d_y) to denote the coordinates of the distance value d. Two stego pixels p'_1 and p'_2 can be calculated by the following equation:

$$\begin{cases} p'_1 = p_1 + d_x \\ p'_2 = p_2 + d_y \end{cases} \quad (4)$$

In the extracting phase, the secret digit can be obtained by calculating the DCV of two stego pixels, that is, $f(p'_1, p'_2)$.

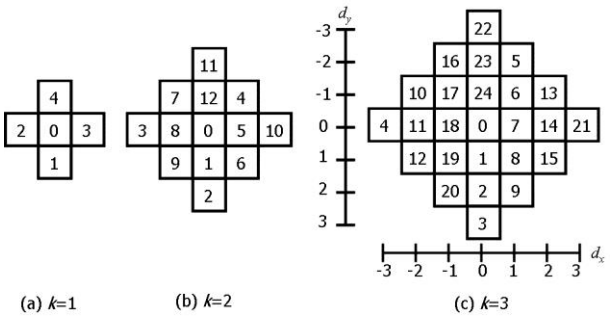


Figure 1: The diamond matrix under different values of k.

2.2 Square Matrix Encoding (SME)

In 2013, Chen et al. proposed their Square Matrix Encoding (SME) [5] to reduce image distortion in the case when k is equal to 3. Through observation, it can be seen that the outermost points (3, 0), (-3, 0), (0, 3), and (0, -3) are far away from the origin of coordinates. This may cause great distortion when the distance value d is 21, 4, 22 or 3. In order to reduce the distortion, the four values are moved to the four corners of the diamond matrix to form a square matrix (SM), as shown in Figure 2.

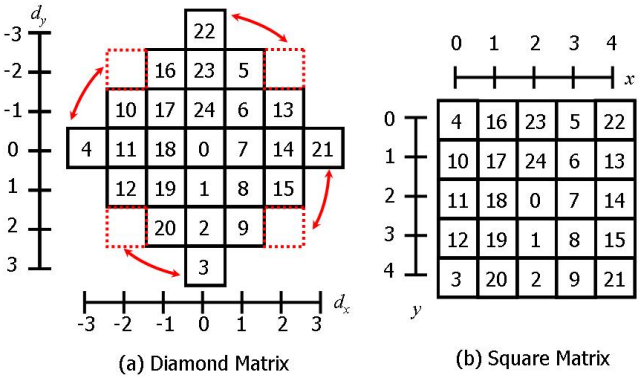


Figure 2: The diamond matrix and the square matrix.

Since the square matrix is obtained, the details of Square Matrix Encoding can be introduced. First of all, the secret data is converted to the secret digit s_B in base B. As for the cover image, two non-overlapping neighboring pixels p_1 and p_2 are taken as a pair to embed the secret digit. The coordinate (x, y) of a cover pixel pair in the square matrix is $(p_1 \bmod 5, p_2 \bmod 5)$. Assume the coordinate (x, y) of the secret digit s_B in the square matrix is (x', y'), that is, $SM(x', y')$ is equal to s_B. The distance between two coordinates can be calculated as:

$$\begin{cases} d_x = x' - (p_1 \bmod 5) \\ d_y = y' - (p_2 \bmod 5) \end{cases} \quad (5)$$

The values d_x and d_y are further modified as:

$$\begin{cases} d'_x = d_x + 5 & \text{if } d_x < -2, \\ d'_x = d_x - 5 & \text{if } d_x > 2, \\ d'_x = d_x & \text{otherwise.} \end{cases} \quad (6)$$

$$\begin{cases} d'_y = d_y + 5 & \text{if } d_y < -2, \\ d'_y = d_y - 5 & \text{if } d_y > 2, \\ d'_y = d_y & \text{otherwise.} \end{cases} \quad (7)$$

Finally, a stego pixel pair p'_1 and p'_2 can be obtained by the following equation:

$$\begin{cases} p'_1 = p_1 + d'_x \\ p'_2 = p_2 + d'_y \end{cases} \quad (8)$$

In the extracting phase, the coordinate (x', y') of a stego pixel pair in the square matrix can be calculated using (p'_1 mod 5, p'_2 mod 5). The secret digit can be found by extracted the digit from $SM(x', y')$.

3 The Proposed Method

Because the procedures for Diamond Encoding and Square Matrix Encodingare in the public domain, and because both produce unique matrices, an attacker can extract the secret data from a stego image, once he/she knows there is something in this stego image. The simplest way to overcome this problem is to encrypt the secret data before embedding. However, an additional burden is necessary. This paper proposes a method to embed secret data which can not be extracted by the attacker. Moreover, the quality of the stego images derived from our proposed method is better than those from both Diamond Encoding and Square Matrix Encoding.

The concept of the proposed method is simple. For a given cover image and secret data, a adjusted-square matrix for them is prodcued. The adjusted-square matrix comes from adjusting the digits in the square matrix. The secret data are embedded into the cover image using the adjusted-square matrix to obtain a stego image. The adjusted-square matrix will be treated as a secret key which is transferred to the receiver through a secure channel. It is obvious that the number of all possible adjusted-square matrices is 25!. It is impossible for an attacker to extract the secret data by trying all possible matrices. Only the person with the secret key can extract the secret data.

Since there are so many adjusted-square matrices, finding out which one is better for a given cover image and secret data is important. It goes without saying that one adjusted-square matrix can be used to produce its corresponding stego image. That also means different adjusted-square matrices create different image qualities of the stego images. If a good adjusted-square matrix can be found, the good quality of the stego image can be achieved. But, how can we single it out?

In this paper, an Adjusted Matrix Encoding (AME) is proposed. There are two main stages in the proposed method. The first stage is to find a good adjusted-square matrix for a given cover image and secret data. The second stage then uses the same procedures as Chen et al.'s Square Matrix Encoding to embed secret data into the cover image. Since the second stage is exactly the same as Chen et al.'s Square Matrix Encoding except that the square matrix is replaced by the adjusted matrix, the details of the second stage are not described redundantly. We focus on Stage 1.

First of all, an adjusted matrix is defined as $D_{5\times5}$= $\{d[i][j] | \ 0 \leqq i, j<5\}$. The positions of all elements in $D_{5\times5}$ are marked from position 0 to 24 sequentially in the way of row major. To obtain a good adjusted-matrix, a square error matrix (*SEM*) is needed first. The *SEM* will help us to find a good adjusted-matrix. The *SEM* is defined as $M_{25\times25}$ $=\{m[i][j] | \ 0 \leqq i, j<25\}$. The content of $m[i][j]$ means the sum of square errors when the digit with value i is put in the position j of the adjusted matrix.

To fill all elements in the *SEM*, the secret data is converted to a numeral base with 25-ary for given secret data. Then, two non-overlapping neighboring pixels of a given cover image are taken as a pair. For each pair (p_1, p_2) and its corresponding secret digit s, the value of $m[s][j]$ for all $0 \leqq j<25$ should be calculated. More precisely, the coordinate (x, y) of a cover pixel pair can be calculated as (p_1 mod 5, p_2 mod 5). Moreover, the coordinate (x', y') of position j in the adjusted matrix can also be calculated as $(\lfloor j/5 \rfloor, j \bmod 5)$. The distances of two coordinates d'_x and d'_y can be obtained through Equation (5) to Equation (7) in Section 2. Therefore, the equation to update $m[s][j]$ is shown given as:

$$m[s][j] = m[s][j] + d'^2_x + d'^2_y. \quad (9)$$

Note that the content of $m[s][j]$ means the sum of square errors when the digit with value s is put in the position j of the adjusted matrix. It is possible for the digit with value s to be put in the position from 0 to 24. That is reason why we must calculate and update all possible $m[s][j]$ where $0 \leqq j<25$ in the *SEM*.

After scanning all cover pixel pairs and secret data, the content of all $m[i][j]$ where $0 \leqq i,j<25$ is filled. All we have to do is to select 25 elements from the *SEM* under the condition that exactly one element is picked up in each row and column, and the sum of selected elements is as small as possible.

Note that $m[s][j]$ means the sum of square errors when the digit with value s is put in position j. If the 25 elements are $m[k][j_k]$ where k =0 to 24 and $m[0][j_0]$+ $m[1][j_1]$+…+ $m[24][j_{24}]$ is the smallest, then putting 1, 2, …, 24 in position j_1, j_2, …, j_{24} of the adjusted matrix can cause the smallest sum of square errors. That means for each $m[k][j_k]$, the content of the adjusted matrix in $d[\lfloor j_k/5 \rfloor][j_k \bmod 5]$ is equal to k. Therefore, if we can select 25 elements to make

the sum of their values smallest under the described condition, the best adjusted-matrix for the given cover image and secret data can be found. It is difficult to find the smallest sum value. Therefore, the proposed method tries to get the sum value as small as possible according to the following steps:

Input: Square Error Matrix

Output: The selected 25 elements

Step 1: Assign all rows and columns as unprocessed rows and columns.

Step 2: For each unprocessed row k, check each unprocessed column to find the repeat times of the smallest value, the column position j_k of the smallest value, and distance value between the smallest value and the second smallest value.

Step 3: Find the row i whose repeat times is the smallest and its distance value is the largest.

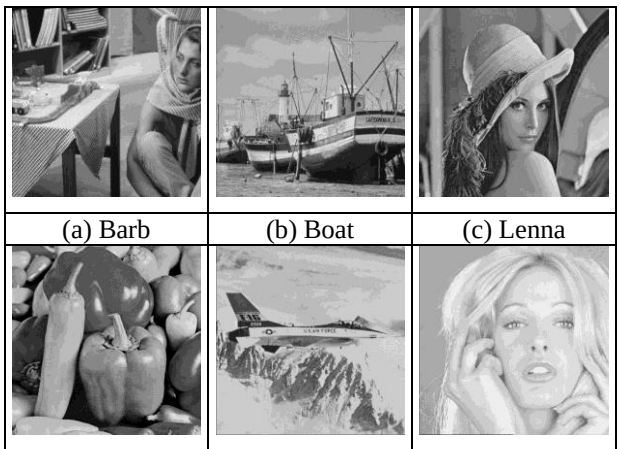
Step 4: Set element $m[i][j_i]$ as the selected element. Mark i row and j_i column as processed row and column.

Step 5: If there are any unprocessed rows, go to Step 2.

Since 25 elements in the *SEM* are selected such that the sum of square errors is as small as possible, the adjusted matrix can be derived according to the elements the above algorithm selected.

4 Experimental Results

The experimental results are 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 3. 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 3. Six 512×512-pixel cover images

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

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

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

$$MSE = \frac{1}{(w \times h)} \sum_{I=1}^{w} \sum_{J=1}^{h} (\alpha(I, J) - \beta(I, J))^2 \quad (11)$$

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 1-3. The values in Table 1 are the average *Square Error* values among different methods when the secret images are embedded into cover images one by one. For comparsion, the distances between the proposed method and other two methods are listed in Table 2. The values in Table 2 come from subtracting the *Square Error* values of the proposed method from those of other two methods. Table 3 shows the average *PSNR* values of three different methods.

Table 1: The average *Square Error* among different methods

Methods	**Cover Images**					
	Barb	*Boat*	*Lenna*	*Pepper*	*Plane*	*Tiffany*
DE[4]	545783	544954	545012	546561	547411	545801
SME[5]	524112	523317	524178	524055	523437	523278
Proposed Method	518232	517835	518331	518176	513307	514293

Table 2. The improvement comparing the proposed method with two methods

Methods	**Cover Images**					
	Barb	*Boat*	*Lenna*	*Pepper*	*Plane*	*Tiffany*
DE[4]	27552	27119	26681	28385	34104	31508
SME[5]	5880	5483	5847	5879	10131	8985

Table 3. The *PSNR* values among different methods

Methods	**Cover Images**					
	Barb	*Boat*	*Lenna*	*Pepper*	*Plane*	*Tiffany*
DE[4]	44.946	44.953	44.952	44.940	44.933	44.946
SME[5]	45.122	45.129	45.121	45.122	45.128	45.129
Proposed Method	45.171	45.174	45.170	45.171	45.212	45.204

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

4 Conclusion

Because the encoding procedures for Diamond Encoding and Square Matrix Encoding are published and the diamond matrix and the square matrix are unique, an attacker can extract the secret data from a stego image if he/she knows there is something in this stego image. In this paper, an adjusted matrix is produced from a given cover image and secret data. It is difficult for an attacker to extract secret data without knowing the adjusted matrix. Moreover, the proposed method adjusted digital data in the matrix to obtain a good adjusted-matrix. Through the good adjusted-matrix, good quality stego images can be derived. According to experimental results, the *Square Error* values of the stego images in the proposed method are always lower than those in the other two methods. All in all, the proposed method is superior to both Diamond Encoding and Square Matrix Encoding.

References

[1] C. S. Chan, "On using LSB matching function for data hiding in pixels," *Fundamenta Informaticae*, vol. 96, pp. 49-59, 2009.

[2] C. K. Chan and L. M. Cheng, " Hiding data in images by simple LSB substitution," *Pattern Recognition*, vol. 37, no. 3, 2004, pp. 469-474.

[3] C. C. Chang, J. Y. Hsiao, and C. S. Chan, "Finding optimal LSB substitution in image hiding by dynamic programming strategy," *Pattern Recognition*, vol. 36, no. 7, pp. 1583-1595, 2003.

[4] R. M. Chao, H. C. Wu, C. C. Lee, and Y. P. Chu, "A novel image data hiding scheme with diamond encoding," *EURASIP Journal on Information Security*, vol. 1-9, Article ID 658047.

[5] J. Chen, C. W. Shiu, and M. C. Wu, "An improvement of diamond encoding using characteristic value positioning and modulus function," *The Journal of Systems and Software*, vol. 86, pp. 1377-1383, 2013.

[6] L. C. Huang, L. Y. Tseng, and M. S. Hwang, "The study of data hiding in medical images," *International Journal of Network Security*, vol. 14, pp. 301-309, 2012.

[7] X. Kang, W. Zeng, and J. Huang, "A multi-band wavelet watermarking scheme," *International Journal of Network Security*, vol. 6, no. 2, pp. 121-126, 2008.

[8] Z. Liao, Y. Huang, and C. Li, "Research on data hiding capacity," *International Journal of Network Security*, vol. 5, pp. 140-144, 2007.

[9] J. Mielikainen, "LSB matching revisited," *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 285-287, 2006.

[10] C. C. Thien, and 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, pp. 2875-2881, 2003.

[11] R. Z. Wang, C. F. Lin, and J. C. Lin, "Hiding by optimal LSB substitution and genetic algorithm," *Pattern Recognition*, vol. 34, pp. 671-683, 2001.

[12] X. Zhang and S. Wang, "Efficient steganographic embedding by exploiting modification direction," *IEEE Communications Letters*, vol. 10, no. 11, pp. 781-783, 2006.

Chi-Shiang Chan was born in Taiwan in 1975. He received the B.S. degree in computer science in 1999 from the National Cheng-Chi University and the M.S. degree in computer science and information engineering in 2001 from the National Chung Cheng University. He received his Ph.D. degree in computer engineering in 2005 also from the National Chung Cheng University. From 2007 to 2010, he has worked as an assistant professor with the Department of Information Science and Applications, Asia University. He is currently an associate professor with the Department of Applied Informatics and Multimedia, Asia University. His research interests include image and signal processing, image compression, information hiding, and data engineering.

Yi-Hui Chen was born in Taiwan in 1979. She received the B.S. and M.S. degrees in information management from Chaoyang University of Technology, in 2001 and 2004, respectively. Afterward, she got her Ph.D. degree in computer science and information engineering from the National Chung Cheng University, in 2009. From 2009 to 2010, she worked with Academia Sinica as a postdoctoral fellow. Later on, she served for IBM Taiwan Collaboratory Research Center as a research scientist. She is now an assistant professor with the Department of Applied Informatics and Multimedia, Asia University. Her research interests include image processing, watermarking, steganography, and XML techniques.

Yuan-Yu Tsai received the B.S. degree in Department of Computer Science and Information Engineering from National Central University, Taiwan, in 2000, and the Ph.D. degree in Institute of Computer Science from National Chung Hsing University, Taiwan, in 2006. He is currently an assistant professor at the Department of Applied Informatics and Multimedia, Asia University, Taichung, Taiwan. His research interests include computer graphics and information hiding algorithms for three-dimensional models and images. He is a member of the ACM and the IEEE Computer Society.