IJCSES International Journal of Computer Sciences and Engineering Systems, Vol. 5, No. 3, July 2011
CSES International © 2011 ISSN 0973-4406

Novel Magic Matrices Generation Method for Secret Messages Embedding

Chin-Chen Chang[1], Kuo-Nan Chen[2] and Huang-Ching Lin[3]

[1]Department of Information Engineering and Computer Science, Feng Chia UniversityTaichung, 40724, Taiwan
E-mail: alan3c@gmail.com

[2]Department of Computer Science and Information Engineering, National Chung Cheng UniversityChiayi, 621, Taiwan, *E-mail: kuonan.chen@gmail.com*

[3]Department of Information Engineering and Computer Science, Feng Chia UniversityTaichung, 40724, Taiwan
E-mail: hata9871@yahoo.com.tw

Abstract: *A flexible data-hiding scheme based on exploiting modification direction (EMD) is proposed in this paper. Our proposed scheme will satisfy objectives users who want to produce high-quality stego images or want a huge hiding capacity. This flexibility is accomplished by a novel magic matrix function developed in our proposed scheme. Our magic matrix function can generate multiple magic matrices with a hiding capacity is decided by users. Then, all the generated magic matrices are tested to find an optimal one that produces the highest-quality stego image. The experimental results show that our proposed scheme outperforms other methods.*

Key words: *Exploiting modification direction, Hiding capacity, Magic matrix*

1. INTRODUCTION

With the rapid improvement of computer and networking technologies, it has become convenient to transmit information via the Internet. However, since the Internet is a public environment, malicious attackers can easily intercept transmitted information. To solve this security problem, many researches are proposed and they can be roughly classified into cryptography systems and data-hiding technologies. Cryptography systems such as AES [1], DES [2], and RSA [3] secure information by transforming the plaintext (original data) to ciphertext by encrypting. Although the ciphertext is difficult to be decrypted without the proper keys, it can attract the attention of malicious attackers because of its apparent meaninglessness. To address this issue, researchers in the data-hiding field aim to embed the secret message in cover file that appears normal.

For digital images, steganography applies data-hiding technology by making the stego image in which the secret message is embedded indistinguishable from the cover image. The simplest and most intuitive way to hide a secret message in cover images is accomplished using the least significant bit (LSB) substitution method [4-6]. However, the exploiting modification direction (EMD) method has better performance than LSB substitution, and it is also the research concerned in this paper. In 2006, Zhang and Wang [7] proposed a steganography scheme in which one secret digit in a $(2n + 1)$-ary notation system can be embedded in n cover pixels, but its hiding capacity was limited to a maximum embedding rate of approximately 1.161 bpp (bits per pixel), where n equals to 2. In 2008, Lee *et al.* [8] proposed an improved EMD scheme in which a cover pixel pair could carry a secret digit in a $(2n + 1)$-ary notation system. The data hiding capacity is kept improving to 2 bpp in [9] proposed by Wang *et al.* in 2010.

We propose a flexible data hiding scheme based on EMD that can satisfy various user objectives in terms of hiding capacity or stego image quality. The content of the paper is organized as follows. Section 2 illustrates the proposed scheme in detail, including preprocessing, embedding, and extracting phases. Section 3 provides the experimental results that support and illustrate the performance of our proposed scheme. Finally, some conclusions are given in Section 4.

Manuscript received August 25, 2010
Manuscript revised December 15, 2010

2. THE PROPOSED SCHEME

Our proposed scheme is an extraordinarily flexible scheme that can satisfy the different objectives of users, whether they need huge hiding capacity or great stego image quality. When the user decides how to utilize the scheme, a number of appropriate magic matrices are generated accordingly. Subsequently, we employ a dynamic programming liked concept to produce the highest-quality stego image available in the magic matrices generated. The proposed scheme is illustrated in three phases: the preprocessing phase, the data-embedding phase, and the data-extracting phase.

2.1. Prepossessing Phase

The prepossessing phase depicts how to generate the appropriate number of magic matrices for later use with data embedding. In our proposed scheme, all cover pixels are first grouped into pixel pairs (X, Y) without overlapping. The secret digits can be embedded in each pixel pair by adjusting its pixel values. For a grey level image C, a magic matrix M, sized $n \times n$, can be generated according to the magic matrix generation function with two parameters, α and β:

$$M(X, Y) = (X \times \alpha + Y \times \beta) \bmod (n^2), \qquad (1)$$

where X and Y are all within [0, 255]. Based on the property of n, the rules for generating magic matrix M with Eq. 1 can be explained using two cases. The corresponding rules for these two cases are shown as follows.

In Case A, n is a prime number:

Rule A1: the two parameters α and β are integers within [1, n^2-1]

Rule A2: $\alpha \neq \beta$

Rule A3: $(\alpha + \beta) \bmod n \neq 0$

Rule A4: gcd (α, n) ≠ 1 or gcd(β, n) ≠ 1, where gcd is the abbreviation of "greatest common divisor"

In Case B, n is a composite number:

Rule B1: the two parameters α and β are integers within [1, n^2-1]

Rule B2: $\alpha \neq \beta$

Rule B3: $(\alpha + \beta) \bmod n \neq 0$

Rule B4: gcd(α, n) ≠ 1 or gcd(β, n) ≠ 1, where gcd is the abbreviation of "greatest common divisor"

Rule B5: $\begin{cases} \alpha \bmod n = 0, \text{and} \\ (\alpha / n) \bmod \gamma \neq 0, \end{cases}$

and

$\begin{cases} \beta \bmod n = 0, \text{and} \\ (\beta / n) \bmod \gamma \neq 0, \end{cases}$ where γ is the set of prime factors of n.

For example, assume n = 3 (Case A), all possible appearances for (α, $\gamma\beta$) are depicted in Table 1.

Table 1
All Possible Appearances of (α, β) when n = 3

(α, β)					
(1, 3)	(3, 1)	(3, 7)	(5, 3)	(6, 4)	(7, 3)
(1, 6)	(3, 2)	(3, 8)	(5, 6)	(6, 5)	(7, 6)
(2, 3)	(3, 4)	(4, 3)	(6, 1)	(6, 7)	(8, 3)
(2, 6)	(3, 5)	(4, 6)	(6, 2)	(6, 8)	(8, 6)

On the other hand, if n = 4 (Case B), all combinations of (α, β) are listed in Table 2.

Table 2
All Possible Appearances of (α, β) when n = 4

(α, β)			
(4, 1)	(4, 9)	(12, 1)	(12, 9)
(4, 3)	(4, 11)	(12, 3)	(12, 11)
(4, 5)	(4, 13)	(12, 5)	(12, 13)
(4, 7)	(4, 15)	(12, 7)	(12, 15)

After the three parameters α, β, and n are decided, the corresponding magic matrix $M(X, Y)$ can be generated by entering X and Y from 0 to 255. For instance, assume n = 3, α = 1, and β = 3; the magic matrix M is created with the following calculations.

$M(0, 0) = (0 \times 1 + 0 \times 3) \bmod (9) = 0,$
$M(0, 1) = (0 \times 1 + 1 \times 3) \bmod (9) = 3,$
$M(0, 2) = (0 \times 1 + 2 \times 3) \bmod (9) = 6,$
$M(1, 0) = (1 \times 1 + 0 \times 3) \bmod (9) = 1,$
$M(1, 1) = (1 \times 1 + 1 \times 3) \bmod (9) = 4,$
$M(1, 2) = (1 \times 1 + 2 \times 3) \bmod (9) = 7,$
$M(2, 0) = (2 \times 1 + 0 \times 3) \bmod (9) = 2,$
$M(2, 1) = (2 \times 1 + 1 \times 3) \bmod (9) = 5,$
$M(2, 2) = (2 \times 1 + 2 \times 3) \bmod (9) = 8.$

The segment of the magic matrix produced is shown as Fig. 1.

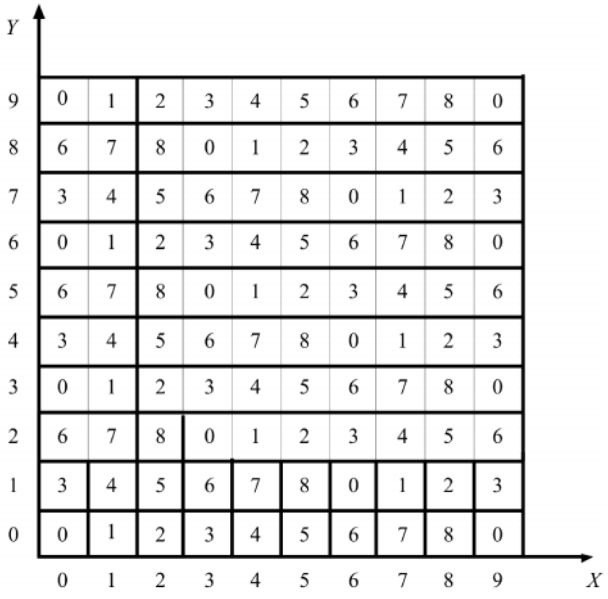


Figure 1: The Segment of Magic Matrix M where $n = 3$, $\alpha = 1$, and $\beta = 3$

Note that the numbers from 0 to 8 appear in all 3×3 blocks of the matrix M shown in Fig. 1.

2.2. Embedding Phase

Before the embedding processes, the secret data is transformed into n^2-ary notation system. Each n^2-ary digit is embedded into one pixel pair of the cover image. Now, we illustrate how to embed the n^2-ary secret digits into pixel pairs. The pixel values are first mapped in the cover pixel pair (P_1, P_2) to the coordinates (X, Y) of M, and a number N. Let N be the center point and select a 3×3 block in M. To embed the corresponding n^2-ary secret digit s, (P_1, P_2) is adjusted to (S_1, S_2), where $M(S_1, S_2) = s$. The embedding flow diagram is shown in Fig. 2.

Consider two examples to trace the embedding procedure.

Example 1: Assume $n = 3$, $\alpha = 1$, and $\beta = 3$, cover pixel pairs are (0, 0), (6, 1), (2, 8), and (8, 5), and secret digits = $(5, 5, 8, 3)_9$. The magic matrix refers to Fig. 1.

1. The stego pixel pair $(S_1, S_2) = (2, 1)$ for embedding $s = 5$ to cover pixel pair (0, 0).
2. The stego pixel pair $(S_1, S_2) = (5, 0)$ for embedding $s = 5$ to cover pixel pair (6, 1).
3. The stego pixel pair $(S_1, S_2) = (2, 8)$ for embedding $s = 8$ to cover pixel pair (2, 8).
4. The stego pixel pair $(S_1, S_2) = (9, 4)$ for embedding $s = 3$ to cover pixel pair (8, 5).

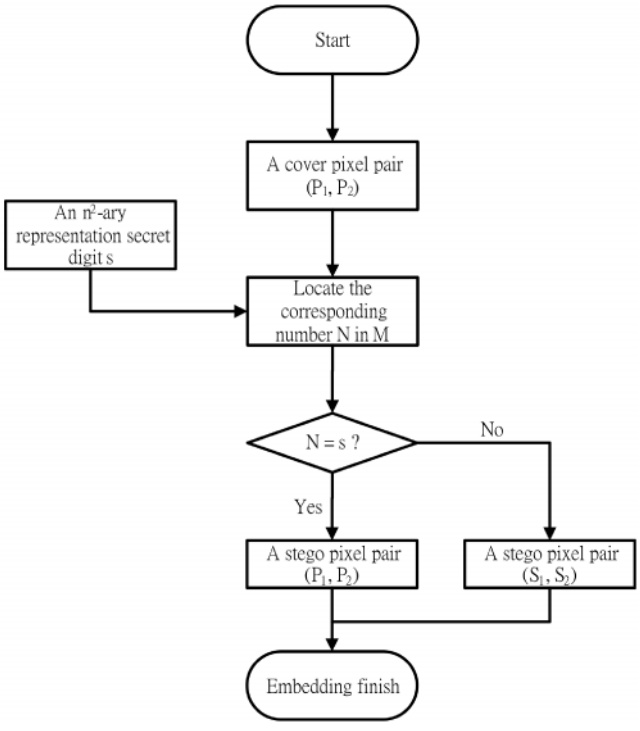


Figure 2: The Embedding Diagram for Embedding an n^2-ary Secret Digit s to Cover Pixel Pair (P_1, P_2)

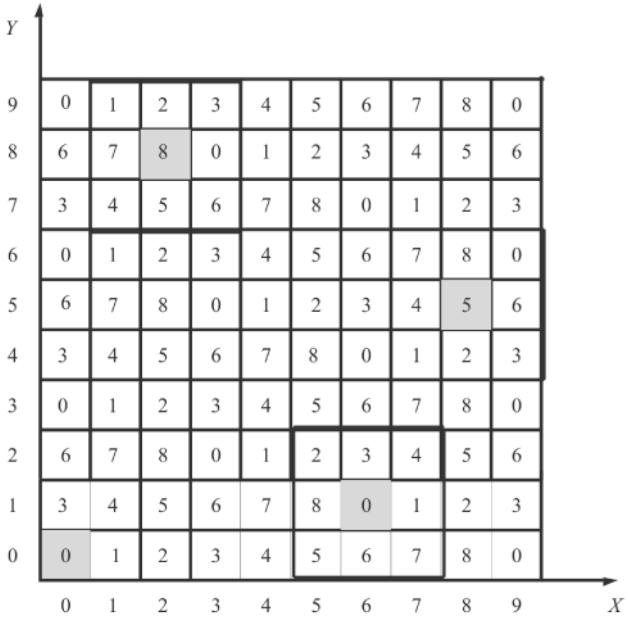


Figure 3: The Magic Matrix in Example 1 (n = 3, $\alpha = 1$, and $\beta = 3$)

Example 2: Assume $n = 4$, $\alpha = 4$, and $\beta = 1$, cover pixel pairs are (0, 0), (7, 0), (1, 6), and (6, 6), and secret digits = $(12, 9, 5, 0)_{16}$. The magic matrix is shown in Fig. 4.

1. The stego pixel pair $(S_1, S_2) = (3, 0)$ for embedding $s = 12$ to cover pixel pair (0, 0).

2. The stego pixel pair (S_1, S_2) = (6, 1) for embedding s = 9 to cover pixel pair (7, 0).
3. The stego pixel pair (S_1, S_2) = (0, 5) for embedding s = 5 to cover pixel pair (1, 6).
4. The stego pixel pair (S_1, S_2) = (6, 8) for embedding s = 0 to cover pixel pair (6, 6).

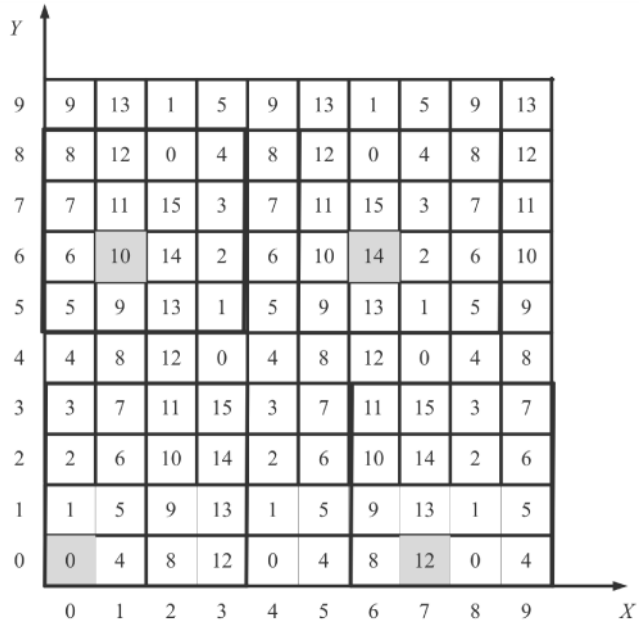


Figure 4: The Magic Matrix in Example 2 (n = 4, α = 4, and β = 1)

Many usable magic matrices can be generated in the preprocessing phase, but only one-the one that results in the best stego image quality based on testing all magic matrices in the embedding phase-is chosen for processing.

2.3. Extracting Phase

At the receiver end, to extract the secret data from the stego image, the receiver needs to produce the magic matrix according to Eq. 1 with parameters n, α, and β, which are transmitted by the sender. Once the magic matrix is generated, the secret digits can be easily extracted in two extracting steps. Step 1 is grouping the stego pixels into pixel pairs without overlapping, and Step 2 is extracting the number N to be secret digit s by mapping each pixel pair to the magic matrix.

We provide take two examples to show the processes of the extracting phase.

Example 3: Following Example 2 in the embedding phase, the stego pixel pairs are (2, 1), (5, 0), (2, 8), and (9, 4). Using the parameters n = 3, α = 1, and β = 3 that the receiver received, the magic matrix can be constructed as shown in Fig. 6. By mapping the stego pixel pairs as the coordinates in the magic matrix, the receiver produces the secret digits' positions as shown in the gray parts in Fig. 6.

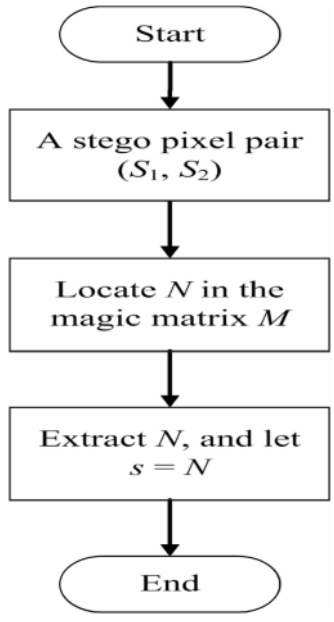


Figure 5: The Extracting Diagram for Extracting an n^2-ary Secret Digit s from Stego Pixel Pair (S_1, S_2)

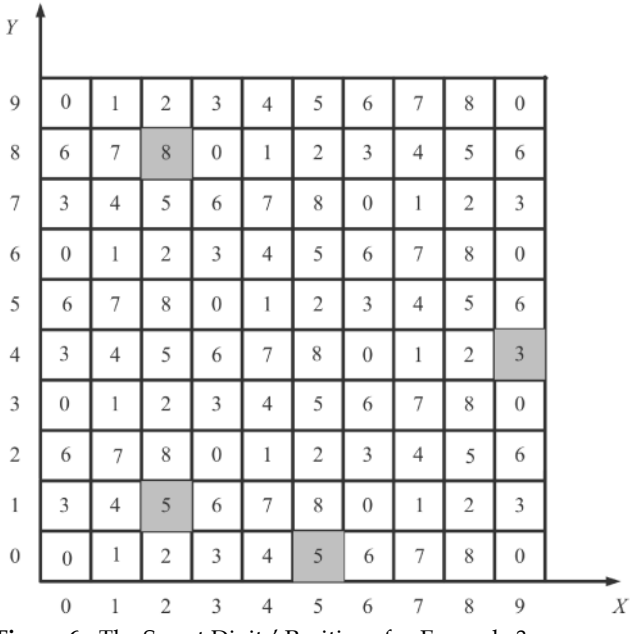


Figure 6: The Secret Digits' Positions for Example 3

Example 4: Following Example 2, the stego pixel pairs are (3, 0), (6, 1), (0, 5), and (6, 8). The receiver can generate the magic matrix by applying the parameters received, i.e., n = 4, α = 4, and β = 1.

The reconstructed magic matrix is shown in Fig. 7. The extracted secret digits are shown as the gray parts in Fig.s 7 by mapping the stego pixel pairs as coordinates in the magic matrix.

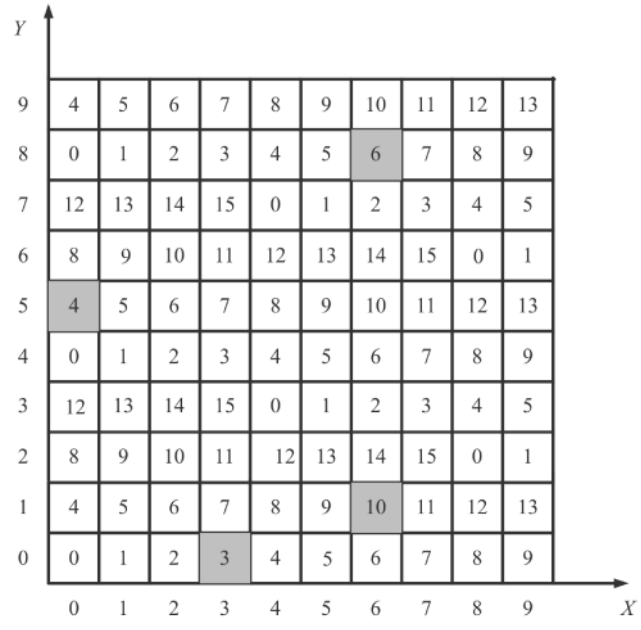


Figure 7: The Extracted Secret Digits' Positions in Example 4

3. EXPERIMENTAL RESULTS

Four gray-level test images, Baboon, Boat, Jet, and Lena, each sized 512×512 pixels, are used in our experiments, as shown in Fig. 8.

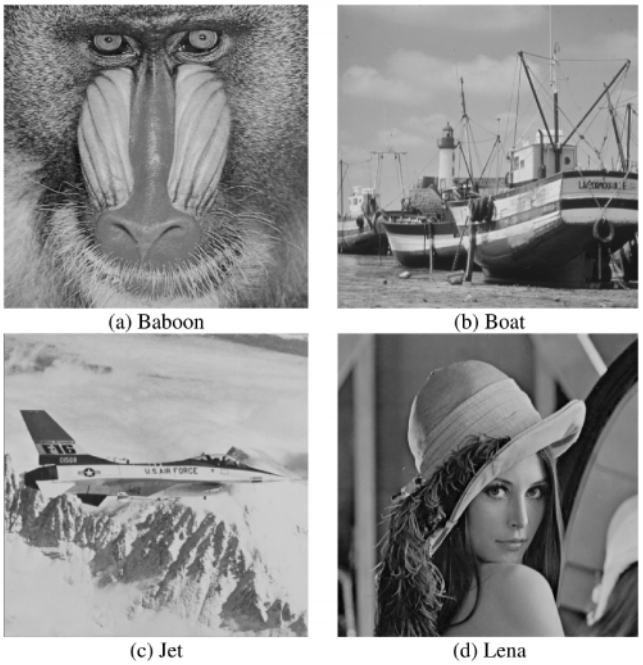


Figure 8: Four Test Images

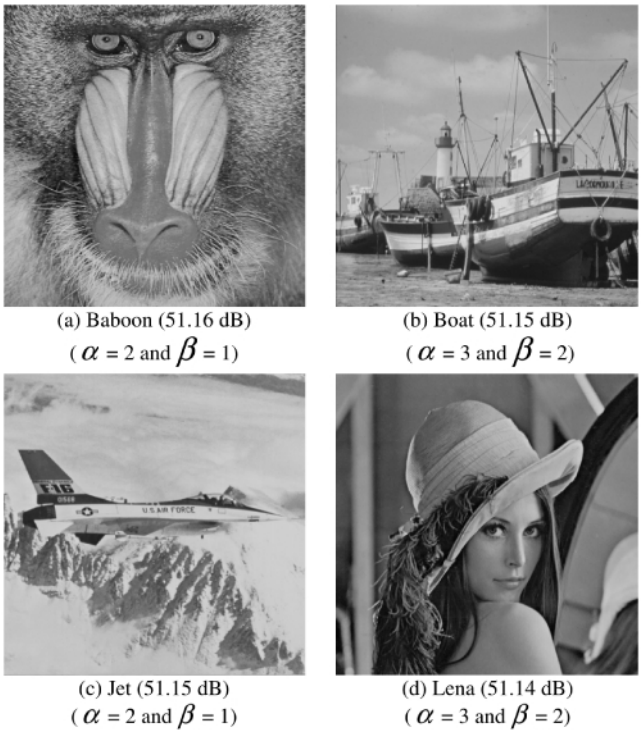


Figure 9: Four Stego Images with PSNR Values by Setting n = 2

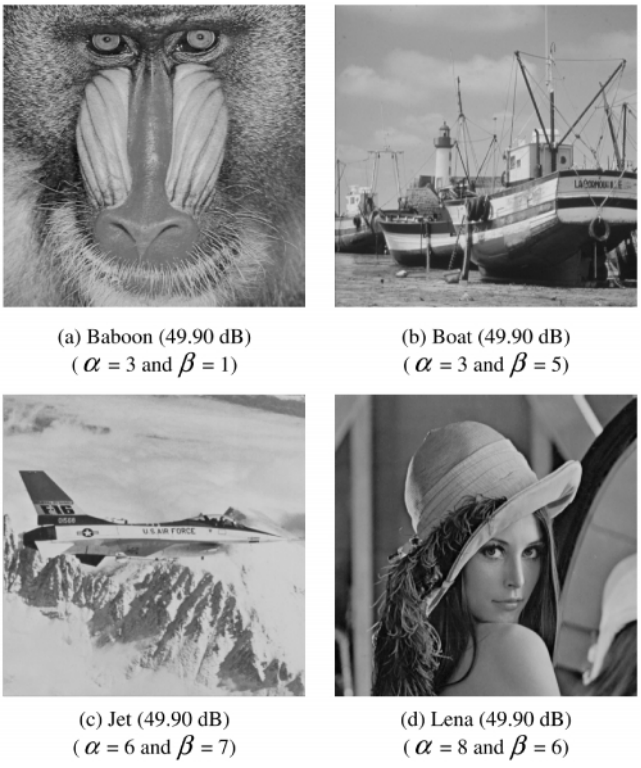
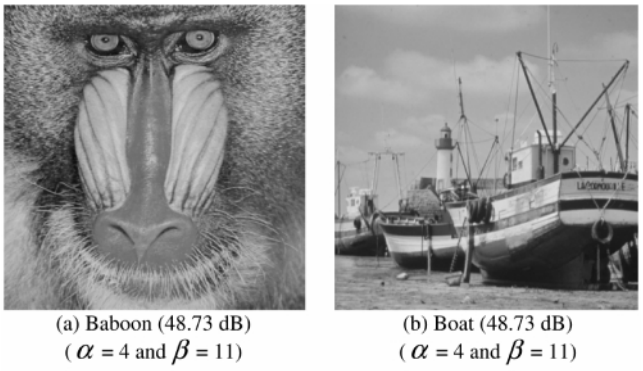


Figure 10: Four Stego Images with PSNR Values by Setting n = 3



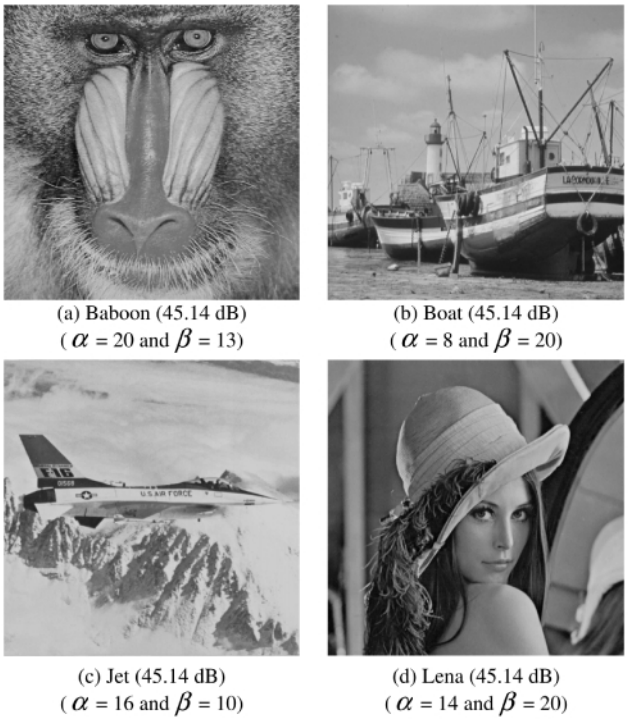
(a) Baboon (48.73 dB) (α = 4 and β = 11)

(b) Boat (48.73 dB) (α = 4 and β = 11)

(c) Jet (48.74 dB) (α = 4 and β = 15)

(d) Lena (48.75 dB) (α = 12 and β = 13)

Figure 11: Four Stego Images with PSNR Values by Setting n = 4



(a) Baboon (45.14 dB) (α = 20 and β = 13)

(b) Boat (45.14 dB) (α = 8 and β = 20)

(c) Jet (45.14 dB) (α = 16 and β = 10)

(d) Lena (45.14 dB) (α = 14 and β = 20)

Figure 12: Four Stego Images with PSNR Values by Setting n = 5

The stego image quality is measured by adapting peak signal-to-noise ratio (PSNR), the formula for which is defined as:

$$\text{PSNR} = 10 \times \log_{10}(255^2/\text{MSE}) \text{ (dB)},$$

where MSE is the mean square error and is used to measure the difference between the cover image and the stego image. The equation for MSE is:

$$\text{MSE} = \sum_{i=1}^{H}\sum_{j=1}^{W}\left(I_{ij} - I'_{ij}\right)^2 / (H \times W),$$

where H and W are the height and width of the cover image, respectively.

To illustrate the performance of our proposed scheme further, we compared our scheme with the scheme proposed in [8] and [9] and list the results of the comparison in Table 3. Table 3 shows that our proposed scheme has better stego image quality than other methods while maintaining similar hiding capacity. In addition, our scheme outperforms others in terms of its flexible hiding capacity.

Table 4 shows the execution time of our proposed scheme using the test image Baboon. The computer environment of the experiment was CPU: Intel Core 2 duo 1.83 GHz; 1 GB main memory, Matlab 2008a, and Windows Vista.

Table 3
Comparisons of Previous Works and Our Proposed Scheme

Test images		*PSNR (dB)* *Baboon*	*Boat*	*Jet*	*Lena*	*Hiding capacity (bpp)*
Lee *et al.* [8]		44.28	44.33	44.45	44.31	1.74
Wang *et al.* [9]		45.15	45.16	46.16	45.16	1.99
Kim *et al.* [10]		49.89	49.88	49.89	49.89	1.58
Ours	n = 2	51.16	51.15	51.15	51.14	1
	n = 3	49.90	49.90	49.90	49.90	1.58
	n = 4	48.73	48.74	48.73	48.75	2
	n = 5	45.14	45.14	45.14	45.14	2.32

Table 4
Execution Time for Baboon

Vaule of n	n = 2	n = 3	n = 4	n = 5
Execuion Time (sec)	2.3154	8.5352	57.3168	93.0391

4. CONCLUSIONS

This paper proposes a flexible data-hiding scheme based on the concept of exploiting modification direction (EMD). We developed a novel magic matrix generation function that can generate various magic matrices by adjusting the parameters so our proposed method is suitable for various

objectives of users who require a high-quality stego image or huge hiding capacity. The performance of our proposed scheme is demonstrated in the experimental results, where it outperforms the previously proposed methods.

REFERENCES

[1] National Institute of Standards & Technology, "Announcing the Advanced Encryption Standard (AES)", *Federal Information Processing Standards Publication*, **197**, 2001.

[2] E. Schaefer, "A Simplified Data Encryption Standard Algorithm", *Cryptologia*, **20**(1), 1996, 77-84.

[3] R. L. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-key Cryptosystems", *Communications of the ACM*, **21**(2), 1978, 120-126.

[4] C. K. Chan, and L. M. Cheng, "Hiding Data in Images by Simple LSB Substitution", *Pattern Recognition,* **37**(3), 2004, 469-47.

[5] C. C. Chen, and C. C. Chang, "LSB-based Steganography using Reflected Gray Code", *IEICE Transactions on Information and Systems*, **E91-D**(4), 2008, 1110-1116.

[6] C. H. Yang, "Inverted Pattern Approach to Improve Image Quality of Information Hiding by LSB Substitution", *Pattern Recognition*, **41**(8), 2008, 2674-2683.

[7] X. Zhang, and S. Wang, "Efficient Steganographic Embedding by Exploiting Modification Direction", *IEEE Communications Letters*, **10**(11), 2006, 781-783.

[8] C. F. Lee, C. C. Chang, and K. H. Wang, "An Improvement of EMD Embedding Method for Large Payloads by Pixel Segmentation Strategy", *Image and Vision Computing*, **26**(12), 2008, 1670-1676.

[9] Z. H. Wang, T. D. Kieu, C. C. Chang, and M. C. Li, "A Novel Information Concealing Method based on Exploiting Modification Direction", *Journal of Information Hiding and Multimedia Signal Processing*, **1**(1), 2010, 1-9.

[10] H. J. Kim, C. Kim, Y. Choi, S. Wang, and X. Zhang, "Improved Modification Direction Methods", *Computers and Mathematics with Applications*, **60**(2), 2010, 319-325.