A Minimal Euclidean Distance Searching Technique for Sudoku Steganography

Wien Hong
Dept. of Information Management
Yu Da College of Business
Miaoli, Taiwan
e-mail: wienhong@ydu.edu.tw

Tung-Shou Chen
Graduate School of Computer Science
and Information Technology
National Taichung Inst. of Technology
Taichung, Taiwan
e-mail: tschen@ntit.edu.tw

Chih-Wei Shiu
Dept. of Information Management
Yu Da College of Business
Miaoli, Taiwan
e-mail: 95104503@ydu.edu.tw

Abstract—**Sudoku, a simple and fun game of logic, has been used for steganography to conceal messages into a digital image recently. Chang et al. adapted the idea of smallest Manhattan distance, embedding secret messages into the neighbors of the located element according to a given Sudoku solution. Hong et al. improved Chang et al.'s technique by introducing additional set of candidate elements to reduce distortions. However, the aforementioned methods suffer from undesirable distortions because the Manhattan distance architectures are used in their method. The proposed method suggests a new scheme for searching embedding positions based on the nearest Euclidean distance, so that minimal distortions can be reached. The experimental results show that, in average, the visual quality of stego image is 1.70 dB higher than that of Chang et al.'s method, and 0.72 dB higher than that of Hong et al.'s method under the same embedding capacity.**

***Keywords**- data hiding; steganography; sudoku*

I. INTRODUCTION

Data hiding is a technique that embeds data into digital media for the purpose of secret communication [1]. Researches of this area have found many applications such as content authentication, covert communication, copyright protection, tamper detection, digital fingerprinting, digital forensics, etc. Digital images are often used as carriers for concealing data because they are often delivered over Internet, and thus may arouse little suspicious when messages are embedded. Data embedding inevitably results in distortion of the original image; therefore, minimizing the distortions caused by data embedding is one of the most important issues in data hiding.

According to the domain of data to be embedded, data hiding technique can mainly be classified into two categories: spatial domain data hiding and frequency domain data hiding. The method of the first type embeds data directly into the pixel values of the host image [2,3,4]. Techniques of this type offer relatively high embedding capacity and easy implementation, and thus draw many researchers' attention to investigate data hiding in this area. The method of the second type converts host images into a transformed domain, and then data are embedded into the transformed coefficients [5,6]. Technique of this type often suffers from higher implementation cost and lower embedding rate than those of spatial domain data hiding.

The well known LSB based data hiding technique embeds secrets in the spatial domain, and can be categorized into two types: LSB replacement technique and LSB matching technique. LSB replacement technique simply replaces the LSBs of a host image by secret messages to conceal data. However, this technique decreases odd pixel values either by one or leaves them unchanged, while even values left unchanged or increased by one. As a result, there exists imbalanced distortions and therefore can be detected by some steganalysis technique [7]. LSB matching technique also modifies the LSBs of a host image for data embedding; however, instead of using simple substitution, LSB matching randomly added or subtracted from the pixel value by one in case the message bit does not match the LSB of the pixel value. As a result, it much difficult for steganographic attacks than that of LSB replacement.

In 2006, Mielikainen proposed a data hiding technique based on LSB matching [8]. His method allows the same amount of payload as LSB matching but with fewer modifications to the host image. Zhang and Wang extended Mielikainen's scheme, in which a $(2n+1)$-ary secret digits is carried by n cover pixels and only one pixel is increased or decreased by one at most [9]. Inspired by Zhang and Wang's works, Chang et al. proposed a data hiding scheme based on Sudoku solutions to modify host pixel values for concealing data [10]. Hong et al. found that some possible embedding positions for smaller distortions in Chang et al.'s method are not fully explored [11]. They modified Chang et al.'s embedding rules by allowing candidate pairs falling into more suitable regions in the reference matrix, so that better embedding positions for smaller distortion can be found. However, the Manhattan architecture used in Hong et al.'s method would downgrade the image quality, since smallest Manhattan does not imply smallest Euclidean distance, and PSNR, the metric for measuring image quality, adopts Euclidean distance as its distance measures.

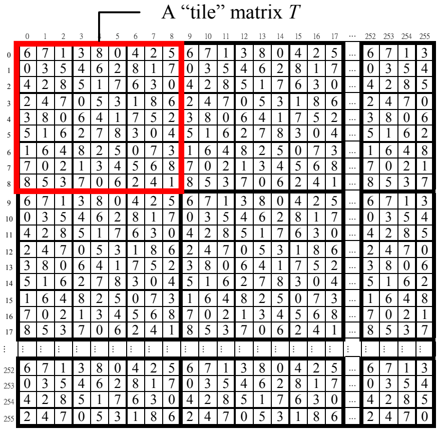
978-0-7695-3494-7/08 $25.00 © 2008 IEEE
DOI 10.1109/ISISE.2008.153

This paper proposes a new element searching scheme for data embedding. Embeddable positions are searched starting from smallest Euclidean distance from the located elements, and then increase the Euclidean distance until embeddable position with smallest the Euclidean distance is found. In his method, the found embeddable position is guaranteed to have the smallest Euclidean distance from the located elements, leading to a higher stego image quality.

II. RELATED WORKS

Sudoku steganography for digital images was first proposed by Chang et al. in 2008. They used a selected Sudoku solution as a reference matrix to guide cover pixels' modification for embedding secret data. To construct a reference matrix *M*, a "tile" matrix *T* is first constructed by subtracting one from a Sudoku solution, so that the digits in matrix *T* ranged from 0 to 8. The reference matrix *M* is then consisting of an $m \times m$ tiling of copies of *T*, where $m = \lfloor 256/9 \rfloor + 1$. Note that for an 8-bit grayscale cover image, matrix *M* has to be truncated to 256×256. An example of a reference matrix *M* for a given Sudoku solution is shown in Figure 1.

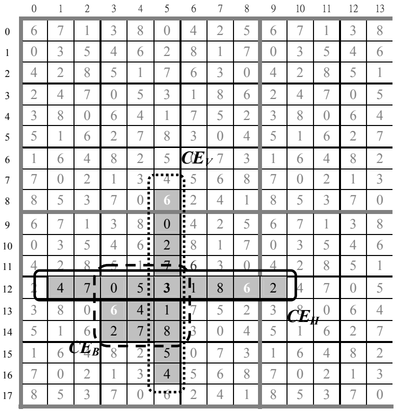
In the data embedding phase, Chang et al. first convert the secret bit stream into digits with base-9 numeral system, and then embed these digits into the cover image. Suppose the converted digits are denoted by $S = s_1 s_2 s_3 \cdots s_n$, where *n* is the total number of converted secret digits and $s_k \in [0,8]$, $1 \le k \le n$. To embed secrete digits, the cover image is partitioned into non-overlapping blocks of size 1×2, and *i*th block consists of a pixel pair (p_{i1}, p_{i2}).


A "tile" matrix T

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	⋯	252	253	254	255
0	6	7	1	3	8	0	4	2	5	6	7	1	3	8	0	4	2	5	⋯	6	7	1	3
1	0	3	5	4	6	2	8	1	7	0	3	5	4	6	2	8	1	7	⋯	0	3	5	4
2	4	2	8	5	1	7	6	3	0	4	2	8	5	1	7	6	3	0	⋯	4	2	8	5
3	2	4	7	0	5	3	1	8	6	2	4	7	0	5	3	1	8	6	⋯	2	4	7	0
4	3	8	0	6	4	1	7	5	2	3	8	0	6	4	1	7	5	2	⋯	3	8	0	6
5	5	1	6	2	7	8	3	0	4	5	1	6	2	7	8	3	0	4	⋯	5	1	6	2
6	1	6	4	8	2	5	0	7	3	1	6	4	8	2	5	0	7	3	⋯	1	6	4	8
7	7	0	2	1	3	4	5	6	8	7	0	2	1	3	4	5	6	8	⋯	7	0	2	1
8	8	5	3	7	0	6	2	4	1	8	5	3	7	0	6	2	4	1	⋯	8	5	3	7
9	6	7	1	3	8	0	4	2	5	6	7	1	3	8	0	4	2	5	⋯	6	7	1	3
10	0	3	5	4	6	2	8	1	7	0	3	5	4	6	2	8	1	7	⋯	0	3	5	4
11	4	2	8	5	1	7	6	3	0	4	2	8	5	1	7	6	3	0	⋯	4	2	8	5
12	2	4	7	0	5	3	1	8	6	2	4	7	0	5	3	1	8	6	⋯	2	4	7	0
13	3	8	0	6	4	1	7	5	2	3	8	0	6	4	1	7	5	2	⋯	3	8	0	6
14	5	1	6	2	7	8	3	0	4	5	1	6	2	7	8	3	0	4	⋯	5	1	6	2
15	1	6	4	8	2	5	0	7	3	1	6	4	8	2	5	0	7	3	⋯	1	6	4	8
16	7	0	2	1	3	4	5	6	8	7	0	2	1	3	4	5	6	8	⋯	7	0	2	1
17	8	5	3	7	0	6	2	4	1	8	5	3	7	0	6	2	4	1	⋯	8	5	3	7
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮		⋮	⋮	⋮	⋮
252	6	7	1	3	8	0	4	2	5	6	7	1	3	8	0	4	2	5	⋯	6	7	1	3
253	0	3	5	4	6	2	8	1	7	0	3	5	4	6	2	8	1	7	⋯	0	3	5	4
254	4	2	8	5	1	7	6	3	0	4	2	8	5	1	7	6	3	0	⋯	4	2	8	5
255	2	4	7	0	5	3	1	8	6	2	4	7	0	5	3	1	8	6	⋯	2	4	7	0

Figure 1. An example of a reference matrix M.

To conceal a base-9 secret digits s_i into block *i* with pixel pair (p_{i1}, p_{i2}), three candidates $M[x_H, y_H]$, $M[x_V, y_V]$ and $M[x_B, y_B]$ are selected respectively from the set of candidate elements *CE_H*, *CE_V* and *CE_B* such that $M[x_H, y_H] = M[x_V, y_V] = M[x_B, y_B] = s_i$. Here, CE_H, CE_V and CE_B are respectively represents the sets of horizontal, vertical and boxed candidate elements, as depicted in Figure 2. Then, the Manhattan distance between (p_{i1}, p_{i2}) and these three candidates are calculated, and the one with the minimum Manhattan distance, denoted by (p'_{i1}, p'_{i2}), are used to replace (p_{i1}, p_{i2}) for concealing a secret digit s_i.



	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	6	7	1	3	8	0	4	2	5	6	7	1	3	8
1	0	3	5	4	6	2	8	1	7	0	3	5	4	6
2	4	2	8	5	1	7	6	3	0	4	2	8	5	1
3	2	4	7	0	5	3	1	8	6	2	4	7	0	5
4	3	8	0	6	4	1	7	5	2	3	8	0	6	4
5	5	1	6	2	7	8	3	0	4	5	1	6	2	7
6	1	6	4	8	2	5		7	3	1	6	4	8	2
7	7	0	2	1	3	4	5	6	8	7	0	2	1	3
8	8	5	3	7	0	6	2	4	1	8	5	3	7	0
9	6	7	1	3	8	0	4	2	5	6	7	1	3	8
10	0	3	5	4	6	2	8	1	7	0	3	5	4	6
11	4	2	8			7	6	3	0	4	2	8	5	1
12	2	4	7	0	5	3	1	8	6	2	4	7	0	5
13	3	8	0	6	4	1	7	5	2	3	8	0	6	4
14	5	1	6	2	7	8	3	0	4	5	1	6	2	7
15	1	6		8	2	5	0	7	3	1	6	4	8	2
16	7	0	2	1	3	4	5	6	8	7	0	2	1	3
17	8	5	3	7	0	6	2	4	1	8	5	3	7	0

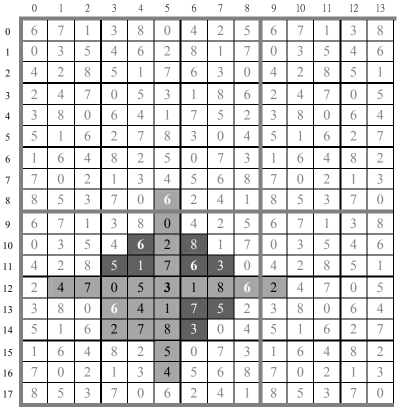
Figure 2. Illustration of CE_H (solid line), CE_V (dotted line) and CE_B (dashed line). The located element $P_i = M[p_{i1}, p_{i2}] = M[12,5] = 3$.

The embedded data can be extracted directly from the stego image by referencing the same Sudoku solution used in the embedding phase. To extract secret digits, stego image is partitioned into non-overlapping blocks of 1×2 pixels using the same technique used in embedding phase. For block *i* with pixel pair (p'_{i1}, p'_{i2}), the secret digit is simply obtained by referencing the reference matrix *M* at the position (p'_{i1}, p'_{i2}), i.e., $s_i = M[p'_{i1}, p'_{i2}]$.

Hong et al. observed that in Chang et al.'s work, some more suitable pairs in the reference matrix M indeed cause smaller distortions; however, these pairs are not included in the sets of candidate pairs in their method. Hong et al. used an additional set of candidate elements CE_A for referencing to reduce distortions. For a located pair (p_{i1}, p_{i2}), CE_A is defined as

$$CE_A = \{(a_{i1}, a_{i2}) \mid (a_{i1}, a_{i2}) \cap (CE_H \cup CE_V \cup CE_B) = \varnothing, \\ D_m((p_{i1}, p_{i2}), (a_{i1}, a_{i2})) < 4\},$$

i.e., CE_A is a set of pairs that disjoint the pairs defined in CE_H, CE_V and CE_B, and their Manhattan distances to the located pair are smaller than 4. Figure 3 depict the pairs belong to CE_A. Note that in Figure 3, either pixel pair (13,3) or (12,8) will be selected for concealing data in Chang et al.'s method, whereas pixel pair (11,6) will be selected in Hong et al.'s method. In this case, the Manhattan distance of Hong's method has been decreased by one, compared to that of Chang et al.'s method.



	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	6	7	1	3	8	0	4	2	5	6	7	1	3	8
1	0	3	5	4	6	2	8	1	7	0	3	5	4	6
2	4	2	8	5	1	7	6	3	0	4	2	8	5	1
3	2	4	7	0	5	3	1	8	6	2	4	7	0	5
4	3	8	0	6	4	1	7	5	2	3	8	0	6	4
5	5	1	6	2	7	8	3	0	4	5	1	6	2	7
6	1	6	4	8	2	5	0	7	3	1	6	4	8	2
7	7	0	2	1	3	4	5	6	8	7	0	2	1	3
8	8	5	3	7	0	6	2	4	1	8	5	3	7	0
9	6	7	1	3	8	0	4	2	5	6	7	1	3	8
10	0	3	5	4	6	2	8	1	7	0	3	5	4	6
11	4	2	8	5	1	7	6	3	0	4	2	8	5	1
12	2	4	7	0	5	3	1	8	6	2	4	7	0	5
13	3	8	0	6	4	1	7	5	2	3	8	0	6	4
14	5	1	6	2	7	8	3	0	4	5	1	6	2	7
15	1	6	4	8	2	5	0	7	3	1	6	4	8	2
16	7	0	2	1	3	4	5	6	8	7	0	2	1	3
17	8	5	3	7	0	6	2	4	1	8	5	3	7	0

Figure 3. Elements defined in CE_A (shaded darker).

III. PROPOSED METHOD

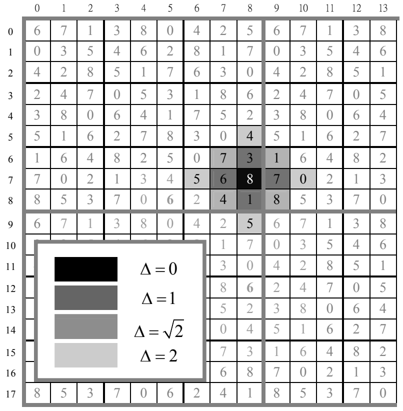
The main idea of the proposed steganographic method is that image quality is evaluated by PSNR, and PSNR uses the square of pixel difference, i.e., the Euclidean distance, as its distance measures. Euclidean distance penalizes large errors much heavier than smaller errors. For example, a single pixel in error by 5 will have the same contribution to PSNR as 25 pixels in error by 1. Therefore, the best candidate for all embeddable pairs should be the pairs with the smallest Euclidean distance.

To minimize the Euclidean distance between the located pair and candidate pairs, a specially designed embedding rule is proposed. Let the located pair be (p_{i1}, p_{i2}) and the set CP_Δ be the pairs with the Euclidean distance Δ to the located pair. Following is the steps to conceal a base-9 digit s_i into pixel pair (p_{i1}, p_{i2}):

Step 1: Set $\Delta = 0$.

Step 2: Scan pixel pairs in CP_Δ sequentially. If the scanned pair $(\hat{p}_{i1}, \hat{p}_{i2})$ satisfying $M[\hat{p}_{i1}, \hat{p}_{i2}] = s_i$, then embedding is done by modifying pixel pair (p_{i1}, p_{i2}) to $(\hat{p}_{i1}, \hat{p}_{i2})$. Otherwise, increase Δ to the next smallest Euclidean distance and repeat Step 2.

Here is an simple example to illustrate the embedding procedures. For example, in Figure 4, suppose the located pair is (7,8), and the secret digits to be concealed is $s_i = 4$. When $\Delta = 0$, there is only one pixel pair (7,8), and $M[7,8] = 8 \neq 4$, therefore, this pair can not embed any data. Now we set $\Delta = 1$. There are four pixel pairs, (7,7), (6,8), (7,9) and (8,8) with $\Delta = 1$, but non of the values of these pairs are equal to 4. For $\Delta = 2$, there are four candidate pairs also, and the pair (8,7) having the value $M[8,7] = 4$, thus, pixel pair (7,8) are modifies to (8,7), and data embedding is done. Note that in Hong et al.'s method, pair (5,8) could possibly be selected to replaced the located pair (7,8). However, the Euclidean distance between pair (5,8) and the located pair (7,8) is 2, the distortion is $\sqrt{2}$ times larger than the proposed method.



	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	6	7	1	3	8	0	4	2	5	6	7	1	3	8
1	0	3	5	4	6	2	8	1	7	0	3	5	4	6
2	4	2	8	5	1	7	6	3	0	4	2	8	5	1
3	2	4	7	0	5	3	1	8	6	2	4	7	0	5
4	3	8	0	6	4	1	7	5	2	3	8	0	6	4
5	5	1	6	2	7	8	3	0	4	5	1	6	2	7
6	1	6	4	8	2	5	0	7	3	1	6	4	8	2
7	7	0	2	1	3	4	5	6	8	7	0	2	1	3
8	8	5	3	7	0	6	2	4	1	8	5	3	7	0
9	6	7	1	3	8	0	4	2	5	6	7	1	3	8
10								1	7	0	3	5	4	6
11								3	0	4	2	8	5	1
12								8	6	2	4	7	0	5
13								5	2	3	8	0	6	4
14								0	4	5	1	6	2	7
15								7	3	1	6	4	8	2
16								6	8	7	0	2	1	3
17	8	5	3	7	0	6	2	4	1	8	5	3	7	0

■ $\Delta = 0$
▨ $\Delta = 1$
▨ $\Delta = \sqrt{2}$
□ $\Delta = 2$

Figure 4. Positions of searching of the proposed method.

IV. EXPERIMENTAL RESULTS

To evaluate the effectiveness of the proposed method, experiments were conducted on four test images, Lena, Jet, Pepper and Baboon, as shown in Figure 5. These test images are also used in Chang et al.'s method, as well as in Hong et al.'s method. The image size is 8-bit each, 512×512 in size. The embedded secret data are generated by PRNG (Pseudo-Random Number Generator). The embedding capacity of the proposed method is $(\log_2 9)/2 \approx 1.585$ bpp, which is the maximum embedding rate under the architecture of 9×9 Sudoku solutions.

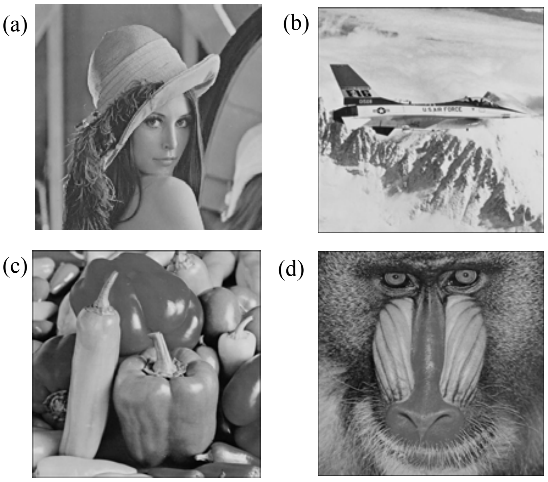


Figure 5. Four test images.

(a) Lena. (b) Jet. (c) Pepper. (d) Baboon.

Since the proposed method selects the pixel pair with the smallest Euclidean distance for data embedding, the selected pixel pair is guaranteed to have the smallest contribution to MSE, and subsequently increases the PSNR.

Table I shows the averaged MSE for all test images. MSE is the mean square error between the original image and the stego image, and is defined as follows:

$$MSE = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left(x_{i,j} - x'_{i,j} \right)^2 ,$$

where $x_{i,j}$ and $x'_{i,j}$ represent the pixel values of the original image and the stego image, respectively. A smaller MSE indicates that the quality of the stego image is closer to the original one. Note that the averaged MSE for our method is 0.882, which is 15.3% lower compared to Hong et al.'s method, and 32.4% lower compared to Chang et al.'s method.

TABLE I. COMPARISON OF MSE IMPROVEMENTS

Images	**MSE**		
	Chang et al.'s	***Hong et al.'s***	***Proposed***
Lena	1.307	1.043	0.883
Jet	1.306	1.037	0.880
Peppers	1.301	1.043	0.884
Baboon	1.301	1.041	0.879
Avg.	1.304	1.041	0.882

Since the averaged MSE of the proposed method is smaller than that of Chang et al.'s and Hong et al.'s method, the proposed method is expected to have the best stego image quality. As can be seen from Table II, in average, the PSNR is 1.70 dB higher than that of Chang et al.'s method, and 0.72 dB higher than that of Hong et al.'s method.

TABLE II. COMPARISON OF STEGO IMAGE QUALITY

Images	**PSNR**		
	Chang et al.'s	***Hong et al.'s***	***Proposed***
Lena	46.98	47.96	48.68
Jet	46.97	47.96	48.68
Peppers	46.98	47.94	48.67
Baboon	46.96	47.93	48.66
Avg.	46.97	47.95	48.67

Besides, the proposed embedding technique scans pixel pairs sequentially by the smallest Euclidean distance to the located pixel. Once the embeddable pixel pair has been found, the embedding procedure for that pixel is done, and no more pixel pairs needed to be scanned. On the other hand, Chang et al.'s work has to scan all the pixels, and has to choose the smallest Manhattan distance among 3 candidate pairs. Hong et al.'s work has to scan all the pixels too, and has to choose the smallest Manhattan distance among 6 candidate pairs. Therefore, as an additional advantage, the proposed method runs slightly faster due to an improved searching technique is used.

V. CONCLUSIONS

In this paper, a revised version of data hiding scheme based on Sudoku solutions is proposed. The proposed method eliminate the major drawbacks of Chang et al.'s and Hong et al.'s method by introducing a new pixel pair scanning order. Pixel pairs with smaller Euclidean distance to the located pixel are scanned first. Therefore, the MSE of the pixel difference can be reduced, leads in higher stego image quality. Compared to previous work, the stego images quality of the proposed method is, in average, 1.71 dB higher than that of Chang et al.'s method, and 0.72 dB higher than that of Hong et al.'s method at the same embedding capacity.

ACKNOWLEDGMENT

This research was supported by the National Science Council of the Republic of China under the Grants NSC97-2622-E-412-002-CC3 and NSC96-2221-E-025-008-MY2.

REFERENCES

[1] F.A.P. Petitcolas, R.J. Anderson, and M.G. Kuhn, "Information Hiding—A Survey," Proceedings of the IEEE, Special Issue on Protection of Multimedia Content, vol. 87, no. 7, pp. 1062-1078, 1999.

[2] C.C. Chang, J.Y. Hsiao and C.S. Chan, "Finding Optimal Least-Significant-Bit Substitution in Image Hiding by Dynamic Programming Strategy," Pattern Recognition, vol. 36, no. 7, pp. 1583-1595, 2003.

[3] C.K. Chan and L.M. Cheng, "Hiding Data in Images by Simple LSB Substitution," Pattern Recognition, vol. 37, no. 3, pp. 469-474, 2004.

[4] D.C. Wu and W.H. Tsai, "A Steganographic Method for Images by Pixel-Value Differencing," Pattern Recognition Letters, vol. 24, no. 9-10, pp. 1613–1626, 2003.

[5] S.D. Lin and C.F. Chen, "A Robust DCT-based Watermarking for Copyright Protection," IEEE Transactions on Consumer Electron, vol. 46, no. 3, pp. 415-421, 2000.

[6] C. C. Chang, and C. Y. Lin, "Reversible Steganography for VQ-compressed Images Using Side Matching and Relocation," IEEE Transactions on Information Forensics and Security, vol. 1, no. 4, pp. 493-501, 2006.

[7] H. Wang and S. Wang, "Cyber Warfare: Steganography vs. Steganalysis," Communications of the ACM, vol. 47, no. 10, 2004.

[8] J. Mielikainen, "LSB Matching Revisited," IEEE Signal Processing Letters, vol. 13, no. 5, pp. 285-287, 2006.

[9] X. Zhang and S. Wang, "Efficient Steganographic Embedding by Exploiting Modification Direction," IEEE Communications Letters, vol. 10, no. 11, pp. 781-783, 2006.

[10] C.C. Chang, Y.C. Chou and T.D. Kieu, "An Information Hiding Scheme Using Sudoku," Proceedings of the Third International Conference on Innovative Computing, Information and Control (ICICIC2008), June 2008.

[11] W. Hong, T.S. Chen and C.W. Shiu, "Steganography Using Sudoku Revisited," to be appeared in International Symposium on Intelligent Information Technology Application, 2008.