2012 Second International Conference on Instrumentation & Measurement, Computer, Communication and Control

A Lossless Secret Image Sharing Scheme based on Steganography

Li Li [a], Ahmed A. Abd El-Latif [b,c], Xuehu Yan [b], Shen Wang [b], Xiamu Niu [a,b,*]

[a] School of Computer Science and Technology, Harbin Institute of Technology Shenzhen Graduate School, Shenzhen 518055, China

[b] School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150080, China

[c] Department of Mathematics, Faculty of Science, Menoufia University, Shebin El-Koom 32511, Egypt

*Corresponding author, Email: xm.niu@hit.edu.cn, Tel: +86-451-86402861

Abstract—**In this paper, a new lossless secret sharing method based on steganography is proposed. The new scheme generates shares from the cover image and secret image, and then embeds the shares into the cover image obtaining stego-shadow images based on 2^4-ary notational system. Experimental results have demonstrated that the proposed secret sharing scheme achieves high quality of the stego-shadow images and high embedding capacity. In addition, it can recover both the original secret image and cover image losslessly.**

***Keywords-** secret sharing; image steganography; lossless recovery*

I. INTRODUCTION

Secret sharing provides a very powerful method by which one secret can be distributed into two or more shares (shadows). When the shares are combined together, the original secret data can be revealed. Blakley as in [1] and Shamir as in [2] independently introduced a secret sharing scheme called a (k, n) threshold scheme. The (k, n) threshold secret sharing mechanism ($k \leq n$) is to partition the secret information into n shares and share them among n participants and needs at least k shares to recover the secret information.

Literature on secret image sharing is quite rich. Herein, we discuss the merits and limitations of some related works and scope of the proposed work. Thien and Lin as in [3] first proposed the secret image sharing method based on Shamir's secret sharing [2] and permutation. It permuted the secret image and performed the secret sharing phase obtaining its shadow images. However, shadow images generated by these kinds of methods [4-7] seem noisy or random which will attract the malicious attacker's attention. To tackle this problem, the secret image sharing based on steganography and Shamir's threshold mechanism has been researched [8-16] where the secret image is hidden in the cover images and meaningful shadow images with little visual change (called stego-shadow images) are generated to avoid the invaders' attention.

In the method proposed by Lin and Tsai [8] and Wu *et al.* [9], the secret image is revealed with distortion. If the secret image is military or medical image, any slight distortion is intolerable. In [10], it improved the work in [9] to restore a distortion free secret image. In addition, the maximum secret capacity of [9, 10] is a quarter of the size of the host image not related to the threshold k. Chang *et al.* [11] shared secret in stego-shadow images with authentication and revealed it lossless based on [3] and Chinese Remainder Theorem (CRT). However, it will cause the expansion of the secret image. It may reduce the capacity of the embedded secret data and distort the quality of stego-shadow images. Lin *et al.* [12] proposed a distortion-free secret image sharing mechanism using modulus operator. Lin and Chan [13] proposed a lossless secret image sharing with steganography based on the quantization. Guo *et al.* [14] proposed a hierarchical threshold secret image sharing based on quantization and Tassa's hierarchical secret sharing which are similar to [10] in the sharing phase. They generated the shares of the secret image and then embedded them into the cover image obtaining stego-shadow images. However, the stego-shadow images do not have high visual quality and embedding capacity, and the cover image could not be reconstructed losslessly. In [15], Chang *et al.* proposed a Sudoku-based secret images sharing with reversibility and Guo *et al.*[16] proposed a secret image sharing based on exploiting modification direction [17]. Both of them, [15, 16], implement the steganography-based image sharing by using (1) with p = 16 and 5 respectively. And s_0 is the information from the cover image and ($s_1, s_2,\ldots, s_{k-1}$) are the values from the secret image in p-ary notational system. For the disguise purpose, the shadow images should be meaningful with satisfactory visual quality. Moreover, the secret image should be recovered without any distortion.

$$f(x) = s_0 + s_1x + s_2*x^2+\ldots+s_{k-1}x^{k-1} \bmod (p). \tag{1}$$

Motivated by the steganography method of Wang's work [18] for embedding the secret digit into cover images with scalable embedding rate (embedding rate: the number of bits that could be embedded in each pixel of the cover image, with unit bits per pixel, i.e. bpp), this paper proposes a new secret image sharing scheme based on 2^4-ary notational system. It generates n stego-shadow images from the secret image and cover image based on Shamir's polynomial and steganography method [18]. Experimental results demonstrate that the validity of secret image sharing with better visual quality than other schemes and has high embedding capacity. Further, it can

978-0-7695-4935-4/12 $26.00 © 2012 IEEE
DOI 10.1109/IMCCC.2012.293

recover both the original secret image and cover image losslessly.

II. The Proposed Scheme

Traditionally, the secret image is shared using the (k-1)-order Shamir's polynomial computation in Galois field GF (p) as in (1) to generate its shadow images. In (1), s_i is one of the pixel values in the secret image in each sharing section, and p is a prime number or 2^m. In the proposed method, $p = 2^{2*2} = 2^4$. Let C denote the cover image with $M_C * N_C$ pixels and S *with* $M_S * N_S$ be the secret image to share. (P_i, P_j) denotes the i-th original pixel pair in C, and (P_i^x, P_j^x) is the i-th pixel pair in x-th stego-shadow image.

A. Stego-shadow images generation phase

The stego-shadow images generation mainly contains the following three procedures.

1) Transform to 2^4-ary representation

Before performing the sharing phase, all pixels of the secret image S are first transformed into 2^4-ary notational system and obtain transformed image $T = \{s_1, s_2,\ldots, s_m\}$ with $m = M_S * N_S * \lceil \log_{2^4} 2^8 \rceil = 2* M_S * N_S$ pixels and $s_q \in [0, 2^4 - 1]$, $q \in [1, m]$.

2) Map cover image pixel into value in the reference matrix

First, each two neighboring pixel values P_i and P_j in cover image C constitute one pixel pair. There are $l = M_C * N_C / 2$ pixel pairs. Then, compute $d_i = F(P_i, P_j)$ by (2) where $d_i \in [0, 2^4 - 1]$ and $i \in [1, l]$.

$$F(P_i, P_j) = (P_i * 2^r + P_j) \bmod 2^{2*r}. \quad (2)$$

3) Stego-shadow image generation

Step 1: Partition the transformed image $T = \{s_1, s_2,\ldots, s_m\}$ into m / (k - 1) number of non-overlapping blocks, each block have k - 1 pixel values.

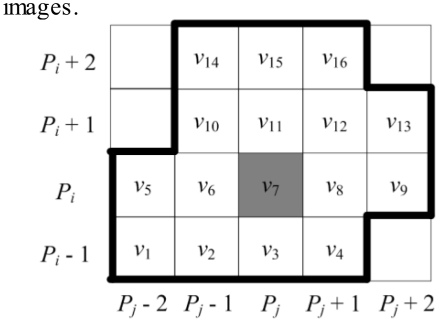
Step 2: For the k - 1 values $s_1^i, s_2^i, \ldots, s_{k-1}^i$ in the block i of T, generate their corresponding n secret shares $f_i(x)$ with x = 1, 2, …, n using (3) where $f_i(x) \in [0, 2^4 - 1]$. The secret shares generation can also be described as the matrix computation over GF(2^4) as in (4). It will generate $f_i(x)$ for all pixel pairs in cover image C with each x.

$$f_i(x) = d_i + s_1^i * x + s_2^i * x^2 + \ldots + s_{k-1}^i * x^{k-1} \bmod (2^4). \quad (3)$$

$$(f_i(1), f_i(2), L\ , f_i(n)) = (d_i, s_1^i, s_2^i, L\ , s_{k-1}^i) * \begin{pmatrix} 1 & 1 & \ldots & 1 \\ 1 & 2 & \ldots & n \\ 1^2 & 2^2 & \ldots & n^2 \\ \ldots & \ldots & \ldots & \ldots \\ 1^{k-1} & 2^{k-1} & \ldots & n^{k-1} \end{pmatrix} (\text{over GF}(2^4)) \quad . \quad (4)$$

Step 3: In the reference block shown in Fig. 1 defined by $v_7 = d_i$, there exists one and only one value equal to $f_i(x)$ satisfying $F(P_i^x, P_j^x) = f_i(x)$ where $P_i^x \in [P_i - 1, P_i + 2]$ and $P_j^x \in [P_j - 2, P_j + 2]$. Thus for each block i of T, compute $(P_i^x, P_j^x) = F^{-1}(f_i(x))$ (x = 1, 2,…, n) which respectively constitute the i-th pixel pair in the *x-th* stego-shadow image SC_x. Each SC_x have the size of $M_C * N_C$.

Step 4: Repeating step 2 to step 3 until all the secret image pixels are embedded into the cover image or the embedded bits achieve the maximum embedding capacity for the definite cover image, the secret image is then hidden in the n stego-shadow images.



	P_j - 2	P_j - 1	P_j	P_j + 1	P_j + 2
P_i + 2		v_{14}	v_{15}	v_{16}	
P_i + 1		v_{10}	v_{11}	v_{12}	v_{13}
P_i	v_5	v_6	v_7	v_8	v_9
P_i - 1	v_1	v_2	v_3	v_4	

Figure 1. The reference block in stego-shadow generation phase of the proposed method

The x-th stego-shadow image will be sent to the x-th participant. When recovering the stego-shadow images, we could obtain both x and $f(x)$ from the x-th participant.

B. Secret recovering phase

Any k of the n stego-shadow images $SC_{x_1}, SC_{x_2}, \ldots, SC_{x_k}$ are used to recover the original secret image S and the cover image C. x_1,…, x_k are k number of distinct values within range [1, n].

1) Secret reveals

Partition each of $SC_{x_1}, SC_{x_2}, \ldots, SC_{x_k}$ into l pixel pairs and map each of them to values in the reference matrix by (2), then we obtain $f_i(x_1)$,…, $f_i(x_k)$ for the i-th pixel pair (P_i, P_j) in the k stego-shadow images, $i \in [1, l]$. After that, solve the k coefficients d_i, s_1^i, s_2^i,…, s_{k-1}^i by performing the matrix computation in Galois field GF(2^4) as in (5).

$$(d_i, s_1^i, s_2^i, L\ , s_{k-1}^i) = (f_i(x_1), f_i(x_2), L\ , f_i(x_k)) * \begin{pmatrix} 1 & 1 & \ldots & 1 \\ x_1 & x_2 & \ldots & x_k \\ x_1^2 & x_2^2 & \ldots & x_k^2 \\ \ldots & \ldots & \ldots & \ldots \\ x_1^{k-1} & x_2^{k-1} & \ldots & x_k^{k-1} \end{pmatrix}^{-1} (\text{over GF}(2^4)) \quad . \quad (5)$$

The recovered $\{s_1, s_2,\ldots, s_m\}$ constructs the secret image in 2^4-ary representation system. Afterwards transform (s_1, s_2,…, s_m) back to the grayscale representation, and recovers the original secret image.

2) Cover image recovery

The above computed d_i ($i \in [1, l]$) and any one of $SC_{x_1}, SC_{x_2}, \ldots, SC_{x_k}$ are used to reveal the cover image. The reference block in Fig. 1 needs some modification to recover the cover image losslessly. For (P_i^x, P_j^x) ($x \in \{x_1, \ldots, x_k\}$), as shown in Fig. 2, the recovery reference block is the region with bold edge determined by (P_i^x, P_j^x) and $v_{10} = f_i(x) = F(P_i^x, P_j^x)$. Here, v_1,…, v_{16} are also 16 distinct values within [0, 15]. Then recover (P_i, P_j) = $F^{-1}(v_q)$ where $v_q = d_i$. After processing all of the pixel pairs in the x-th stego-shadow image, the cover image is recovered without any distortion.

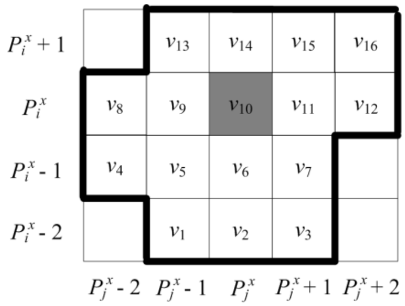


Figure 2. The reference block in secret revealing phase in the proposed method

3) Edge case

When the cover image pixel value is 0, 1, 254 or 255, it will cause underflow or overflow (pixel value < 0 or > 255) while embedding secret image, which is called an edge case. The edge case is treated as follows.

Before the embedding phase, perform the preprocessing according to (6) where p_{edge} is the pixel value in edge case, compute *num* (the amount of p_{edge}, with definite length), and record p_{edge}'s position (x_i, y_i) in the cover image. *Num* and (x_i, y_i) are treated as the indicated information. They are embedded from right bottom to left upper in the order of (*Num*, (x_1, y_1),…(x_{num}, y_{num}) while the secret image is embedded from left upper to right bottom in the cover image. *Num*, (x_i, y_i), and secret image pixels are all embedded using the phases stated in section 3.1.

$$P=\begin{cases} p_{edge}+2, & \text{if } p_{edge}=0,1 \\ p_{edge}-2, & \text{if } p_{edge}=254,255 \end{cases}. \tag{6}$$

In the recovery phase, first follow the secret recovering phase to reconstruct secret image R_S and cover image R_C. After that, obtain *num* beginning from the last pixel pair of R_C, and then have (x_1, y_1),…,(x_{num}, y_{num}) with the same order as in the embedding phase. According to the obtained (x_i, y_i), recover their corresponding cover image pixels p_{cover} by (7).

$$P'=\begin{cases} p_{\text{cover}}-2, & \text{if } p_{\text{cover}}=2,3 \\ p_{\text{cover}}+2, & \text{if } p_{\text{cover}}=252,253 \end{cases}. \tag{7}$$

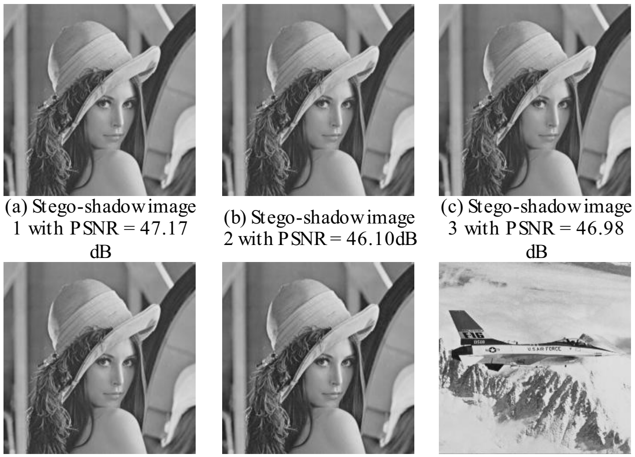
III. EXPERIMENTAL RESULTS AND ANALYSIS

Herein, we conduct experiments and analysis to evaluate the effectiveness of the proposed scheme. All the experiments were done by Matlab7.0 in a computer of Intel Core i3 CPU@2.93 GHz and 3.36 GB of RAM. Several images, standard with size 512×512, are used as the cover images one of them is shown in Fig. 3(a) and F16 as shown in Fig. 3(b), with size 256×256, is the secret image to share. It's noted that, *num* with 16 bits is suitable for the cover image with size 512×512 in the experiment.

Fig. 4 (a-e) shows the four stego-shadow images of *Lena*, and the recovered *Lena*. In the experiment, there are 524272 bits in the secret image with size 256×256 could be embedded into the cover image with size 512×512 when k = 2, Fig.4 (f) shows that the embedded bits could be recovered without any disturb.

(a) Lena (b) F16

Figure 3. Cover image Lena and the secret image F16



(a) Stego-shadow image 1 with PSNR = 47.17 dB

(b) Stego-shadow image 2 with PSNR = 46.10dB

(c) Stego-shadow image 3 with PSNR = 46.98 dB

(d) Stego-shadow image 4 with PSNR = 47.12 dB

(e) The recovered lossless cover image

(f) The extracted secret image with 524272bits

Figure 4. The results of *Lena*, k = 2 and n = 4

The peak signal-to-noise ratio (PSNR), as in (8), is used to measure the quality of the stego-shadow images after embedding secret image where *MSE*, (9), is used to measure the mean square error between the cover image and stego-shadow image. In (9), $u'(x, y)$ is the pixel value of the stego-shadow image corresponding to the pixel value $u(x, y)$ in the cover image.

$$PSNR=10*\log_{10}(\frac{255^2}{MSE})\text{dB}. \tag{8}$$

$$MSE=\frac{1}{M_C\times N_C}\sum_{x=1}^{M_C}\sum_{y=1}^{N_C}[u'(x,y)-u(x,y)]^2. \tag{9}$$

For all of the four stego-shadow images of the fifteen test cover images with threshold setting k = 2 and n = 4, Fig. 5 shows the average PSNR curves comparison among the proposed method, Sudoku-based method [15] and EMD-based method [16]. It turns out that the proposed method has better visual quality. As can be seen from Fig. 4 (a-d) and Fig. 5, the proposed method has high quality from both human vision and objective measurement.

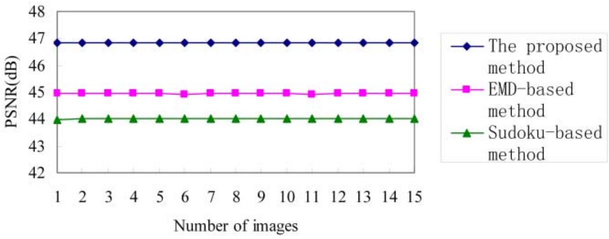


Figure 5. The average PSNR curves for the four stego-shadow images using different methods with 15 test images, $k = 2$, $n = 4$

Table I shows the maximum embedded bits and average embedding capacity for 15 test cover images with n= 4, and k = 2, 3. Sudoku-based method has the largest maximum capacity, a little larger than the proposed method, while EMD-based has the lowest capacity. Then for secret grayscale image with size 256×256, if k = 2, it should choose a little larger image to embed all of the pixels by using the proposed method. But it will embed all of the pixels if k = 3. As observed during the tests, the appropriate cover image size should be considered under definite k and given secret image.

The performance among the proposed method, Sudoku-based method [15] and EMD-based method [16] are summarized in Table II. It shows that the proposed method has the best overall performance in terms of visual quality and embedding rate.

TABLE I. THE CAPACITY (BITS) FOR 15 TEST COVER IMAGES WITH N = 4 AND K = 2, 3

Capacity(bits)	**Average capacity**	**Maximum capacity**	**Average capacity**	**Maximum capacity**
Threshold	k = 2		k =3	
Proposed method	518964	524272	1043251	1048560
Sudoku-based	524288	524288	1048576	1048576
EMD-based	262144	262144	524288	524288

TABLE II. COMPARISON BETWEEN PROPOSED METHOD, SUDOKU-BASED METHOD [15] AND EMD-BASED METHOD [16]

	PSNR (dB) (k = 2)	**Embedding rate (bpp)**
Proposed method	46.75	2(k-1)(1-S / (M * N))
Sudoku-based	44.15	2(k-1)
EMD-based	45.12	k -1

IV. CONCLUSIONS

In this paper, we have presented a new lossless secret image sharing based on steganography. In the new scheme, the steganography method based on 2^4-ary representation system is employed to generate shared data, which are embedded into cover image to form stego-shadow images. By recovering phase, both secret image and cover image can be reconstructed without distortion. Experiments show the superior performance of the proposed scheme to other schemes in terms of visual quality and embedding rate.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China (60832010, 61100187), the Fundamental Research Funds for the Central Universities (Grant No. HIT. NSRIF. 2010046).

REFERENCES

[1] G. R. Blakley, "Safeguarding cryptographic keys," In: Proceedings AFIPS 1979 National Computer Conference, vol. 48,, pp. 313−317, June 1979.

[2] A. Shamir, "How to share a secret. Commun," Communications of the ACM, vol. 22, no.11, pp. 612–613, 1979.

[3] C.C. Thien and J. C. Lin, "Secret image sharing," Comput Graph,vol. 26, no. 5, pp.765–770, 2002.

[4] C. C. Thien and J. C. Lin, "An image-sharing method with user-friendly shadow images," IEEE Trans. on Circuits and systems for video technology,;vol. 13, no. 12, pp.1161-1169, 2003.

[5] R. Z. Wang and C. H. Su, "Secret image sharing with smaller shadow images," Pattern Recogn Lett,vol. 27, no. 6, pp. 551-555, 2006.

[6] R. Zhao, J. J. Zhao, F. Dai, and F. Q. Zhao, "A new image secret sharing scheme to identify cheaters," Computer Standards & Interfaces, vol. 31, no. 1, pp, 252-257, 2009.

[7] L. Li, A. A. Abd El-Latif, Z. F. Shi, and X. M. Niu, "A new loss-tolerant image encryption scheme based on secret sharing and two chaotic systems," Research Journal of Applied Sciences, Engineering and Technology, Maxwell Scientific Organization, vol, 4, no.8, pp. 877-883, 2012.

[8] C. C. Lin and W. H. Tsai, "Secret image sharing with steganography and authentication," The Journal of Systems and Software,vol. 73, no. 3, pp. 405-414, 2004.

[9] Y. S. Wu, C. C. Thien, and J. C. Lin, "Sharing and hiding secret images with size constraint," Pattern Recognition,vol. 37, no. 7,pp. 1377–1385, 2004.

[10] C. N. Yang, T. S. Chen, K. H. Yu, and C. C. Wang, "Improvements of image sharing with steganography and authentication," The Journal of Systems and Software, vol. 80, no. 7, pp. 1070-1076, 2007.

[11] C. C. Chang, Y. P. Hsieh, and C. H. Lin, "Sharing secrets in stego images with authentication," Pattern Recognition,vol. 41, no. 10, pp. 3130-3137, 2008.

[12] P. Y. Lin, J. S. Lee, and C. C. Chang, "Distortion-free secret images haring mechanism using modulus operator," Pattern Recogn,vol. 42, no. 5, pp. 886-895, 2009.

[13] P. Y. Lin and C. S. Chan, "Invertible secret image sharing with steganography," Pattern Recognition Letters, vol. 31, no. 13, pp. 1887-1893, 2010.

[14] C. Guo, C. C. Chang, and C. Qin, "A hierarchical threshold secret image sharing," Pattern Recognition Letters, vol. 33, no. 1, pp. 83-91, 2012.

[15] C. C. Chang, P. Y. Lin, Z. H. Wang, and M. C. Li, "A Sudoku-based secret image sharing scheme with reversibility (Invited paper)," Journal of Communications, vol.5, no. 1, pp. 5-12, 2010.

[16] C. Guo, Z. H. Wang, C. C. Chang, and C. Qin, "A secret image sharing scheme with high quality shadows based on exploiting modification direction," Journal of Multimedia, vol. 6, no. 4, pp. 341-348, 2011

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

[18] D. Wang, "Research of gray-Image steganography based on LSB," Harbin Institute of Technology, Master thesis, December, pp. 16-21, 2008.