http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (7): 1299-1308, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.1299.1308

A Novel Data Embedding Method Using Random Pixels Selecting

[1]Ling Liu, [2]Tungshou Chen, [1]Chen Cao, [1]Xuan Wen and [1]Rongsheng Xie
[1]Faculty of Computer Science and Technology, Xiamen University of Technology,
Xiamen 361024, China
[2]Faculty of Computer Science and Information Engineering,
National Taichung University of Science and Technology, Taichung, Taiwan

Abstract: Data hiding is an effective technique in multimedia security. With regard to the fact that when carrying information, the data hiding algorithm was liable to be detected by statistical steganalysis tool, a novel data embedding method using random pixels selecting was proposed. This method embedded data by using multilevel histogram shifting technique. In the embedding process, random pixels in natural images were selected to be used to hide data. Therefore, the embedded data distributed in a more irregular manner and can better evade the detection of statistical steganalysis tool. Moreover, the proposed method obtained better stego image quality. In comparison to another similar work, the proposed method provided better security while offering low distortion.

Key words: Random pixels selecting, data hiding, subtractive pixel adjacency matrix, histogram modification

INTRODUCTION

Data hiding aims to embed undetectable data into the image, audio, video and other digital media. Especially for military and national security, etc., where the security requirements for data hiding technique is rising, the hidden secret data cannot be easily detected, stego image cannot be distorted and can be recovered to the cover image after extracting the embedded message (Hong and Chen, 2011). Reversible data hiding technique has been extensively developed as it meets above requirements.

According to embedding mode, reversible data hiding methods can be classified into three main categories: Namely, data compression (Hong *et al*., 2011), difference expansion (Tian, 2003) and histogram shifting (Ni *et al*., 2006; Hong *et al*., 2010; Pan *et al*., 2011; Kim *et al*., 2009; Li *et al*., 2010; Luo *et al*., 2011; Zhao and Luo, 2012). The principle of difference expansion is to expand the difference of a pair of pixels to embed secret data. Ni *et al*. (2006) applied the histogram shifting technique to data hiding earlier. In Ni's method, secret data is hidden in the empty bins generated by shifting the histogram. As only one gray value can be modified for each pixel, high-quality stego-image can be obtained. However, as the embedding capacity is influenced by the peak height of the image histogram, the total capacity obtained is relatively low. Hong and Chen (2010) enhanced the performance of embedded algorithm by histogram shifting. Kim *et al*. (2009) proposed a reversible data hiding method based on the modification of difference histograms among sub-sampled images. This method exploited high spatial correlation among sub-sampled images to achieve high embedding capacity. One sub-sampled image is selected as the reference sub-sampled image and then multi-level histogram shifting technique is employed during the data embedding. As more difference modification is allowed this method obtained better embedding capacity. A reversible data hiding method based on the differences modification of neighboring pixels was proposed by Li *et al*. (2010). This method adopted the inverse "S" scan order to hide data in data embedding process and achieved a good embedding capacity. Recently, Luo *et al*. (2011) extended Kim's study and introduced a reversible data hiding scheme on the basis of sub-sampling principle. In this method, the central pixel of each sub-sampled image is selected as a reference and multi-level histogram shifting technique is employed when embedding secret data. In decoder, the cover image can be recovered and the secret data can be extracted. Zhao and Luo (2012) applied the multi-level histogram shifting technique to data hiding and proposed a reversible data hiding method based on "Hilbert curve" scan and histogram modification. In the data embedding process,

Corresponding Author: Ling Liu, Faculty of Computer Science and Technology, Xiamen University of Technology, Xiamen 361024, China

the scan order of pixels is made in the "Hilbert curve" scan. This method had good performance in both stego image quality and embedding capacity.

As mentioned in literature (Kim *et al*., 2009; Li *et al*., 2010; Luo *et al*., 2011; Zhao and Luo, 2012), the data embedding method adopted the sequential scan, the reverse "S" scan or "Hilbert curve" scan during the embedding process. Therefore, hidden data is liable to be detected by detection tools as data is embedded in a certain order. This study proposes a reversible randomly embedded anti-SPAM detection data hiding method. It used random pixels selecting instead of traditional order of image pixel scan to data hiding in natural images. In the data embedding procedure, pseudo-random sequences generated by the random function are applied to mark the unit of data to be hidden and then secret data is hidden into image with the multi-level histogram modification technique. In data extraction, the cover image can be recovered and the hidden data can be extracted. The purpose of this method lied in improving the security of data hiding method against the detection of SPAM and at the time, assuring the stego image quality.

LITERATURE REVIEW

Sub-sampled image and reference sub-sampled image: Sub-sampling (Kim *et al*., 2009) is the process for selecting a series of pixel units from an image. Suppose an image I of size M×N pixels is denoted by $I_{i,j}$, where $I_{i,j} \in G$, i = 0,..., M-1, j = 0,..., N-1, G = {0, 1, 2,..., 255} and two sampling factors Δu and Δv, which refer to sampling intervals along the horizontal direction and vertical direction of the image, respectively. Sub-sampled image S_k is obtained after sub-sampling and can be calculated with the Eq. 1. Figure 1is the sub-sampling process of an image when sub-sampling intervals, Δu and Δv are set to 2:

$$S_k(i,j) = I(i \cdot \Delta v + \text{floor}(\frac{k-1}{\Delta u}),\ j \cdot \Delta u + ((k-1) \bmod \Delta u)) \quad (1)$$

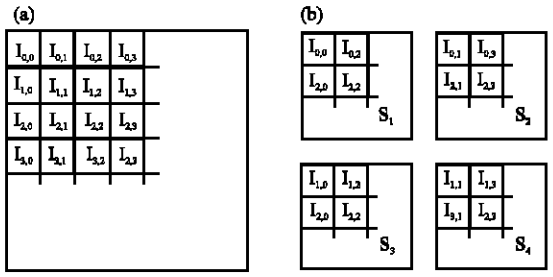

Fig. 1(a-b): Sub-sampling example at Δu = Δv = 2 (a) Original image and (b) Sub-sampled images

Reference sub-sampled image (Kim *et al*., 2009) is selected from the obtained sub-sampled images, which makes the spatial correlation among sub-sampled images the maximum. It is expressed with S_{ref} and can be obtained according to Eq. 2. For instance, reference sub-sampled image in Fig. 1 is S_1:

$$S_{ref} = (\text{Round}(\frac{\Delta u}{2} - 1)) \times \Delta v + \text{Round}(\frac{\Delta v}{2}) \quad (2)$$

SPAM steganalyzer: Subtractive Pixel Adjacency Matrix (SPAM) (Pevny *et al*., 2010) is a modern steganalysis method for detecting stego images with independent random stego signal. It obtains the features of images by calculating the transition probabilities along eight directions and the following is the detailed process.

Suppose a gray image I, the differences between each pixel and eight neighboring pixels in the image are first calculated. The eight neighboring directions are respectively up, down, left, right, upper left, upper right, lower left and lower right. In order to describe the calculation of features in each direction, these specified directions are expressed with such superscripts as (↑, ↓, ←, →, ↖, ↗, ↙, ↘). Then, image features are described with the Markov transition probability matrix, i.e., obtain image features by calculating the conditional probability for differences between each pixel and pixels on eight neighboring directions. The number of features is determined by the range of differences T and the order of Markov chain. The latter is the amount of step adopted by the transition probability, also known as the SPAM order. The detailed calculation of SPAM features is as follows:

Step 1: Calculate the difference array $D^{\bullet}$, e.g., from left to right in horizontal direction:

$$D^{\rightarrow}_{i,j} = I_{i,j} - I_{i,j+1},\ i \in \{0,...,M-1\},\ j \in \{0,..., N-2\} \quad (3)$$

Step 2: Calculate SPAM features of two conditional probabilities, the first-order SPAM features (F^{1st}) and second-order SPAM features (F^{2nd}). The first-order SPAM features (F^{1st}) are constructed by the difference arrays model of first-order Markov process. Take the horizontal direction for example:

$$M^{\rightarrow}_{u,v} = P_r(D^{\rightarrow}_{i,j+1} = u \mid D^{\rightarrow}_{i,j} = v),\ u, v \in \{-T,..., T\} \quad (4)$$

The second-order SPAM features (F^{2nd}) are constructed by the difference arrays model of second-order Markov process. Take the horizontal direction for example:

$$M^{\rightarrow}_{u,v,w} = P_r\,(D^{\rightarrow}_{i,j+2} = u \mid D^{\rightarrow}_{i,j+1} = v,\ D^{\rightarrow}_{i,j} = w) \quad (5)$$

Step 3: Feature sets, F^{1st} and F^{2nd}, finally formed with the horizontal, vertical and diagonal feature matrices are:

$$F^{\bullet}_{1,\ldots,k} = \frac{1}{4}[M^{\rightarrow} + M^{\leftarrow} + M^{\downarrow} + M^{\uparrow}] \quad (6)$$

$$F^{\bullet}_{k+1,\ldots,2k} = \frac{1}{4}[M^{\searrow} + M^{\nwarrow} + M^{\swarrow} + M^{\nearrow}] \quad (7)$$

PROPOSED METHOD

Generally, data embedding is achieved by scanning the pixel to be embedded sequentially or following the order of a certain regular curve. Therefore, detection tools are easier to find out the regularity of embedded data and it is difficult for data hiding methods against the detection. In this study, a reversible data hiding method based on randomized embedding is proposed. The cover image will be first executed the sub-sampling and the difference images among sub-sampled images are constructed. Then the pixels in the difference images are scrambled and secret data would be embedded in such scrambled images by multi-level histogram shifting technique. Finally, the pixels in scrambled difference images are recovered back to the original order and the stego image could be obtained with the reference sub-sampled image. The data extraction and image recovery can be completed by the inverse of the embedding procedure.

Image scrambling: Suppose an image I is M×N pixels in size and scan every pixel from top to bottom and from left to right to get result $P_1, P_2, \ldots, P_{M\times N-1}\ P_{M\times N}$. Scramble the image with a pseudo random sequence $(a_1,a_2,\ldots,a_{M\times N})$, i.e., the order of the sequence is the position of pixels in the scrambled image and then scan each pixel again to get the result $P_{a_1}P_{a_2}\ldots P_{a_{M\times N}}$. Figure 2 gives the scrambling process of a 3×3 image with the pseudo-random sequence of $(a_1,a_2,\ldots,a_9) = (2, 1, 4, 3, 8, 7, 6, 5, 9)$.

Data embedding process:

- **Input:** Cover image I, secret bit string w, sub-sampling intervals Δu and Δv, embedding level L and random sequence
- **Output:** Stego image I′ and overhead information O_{info}

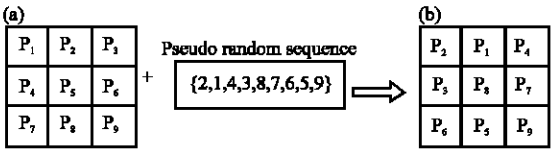

Fig. 2(a-b): Scrambling process of a 3×3 image (a) Original image and (b) Scrambled image

Step 1: According to Eq. 1, execute the sub-sampling of the cover image I using two sub-sampling factors Δu and Δv to obtain a series of sub-sampled images $S_k(i,j)$, where k = 1,..., Δu×Δv. Select the reference sub-sampled image S_{ref} with Eq. 2 to construct difference images E_k between the reference S_{ref} and the other sub-sampled images S_k, k = 1, 2, ..., ref-1, ref+1, ..., Δu×Δv

Step 2: Scramble each difference image E_k and obtain the scrambled difference images E_{RK}. Pixel in E_{Rk} is denoted by e_R. First, set the number of pixels in E_k to C. As for each difference image E_k, a pseudo-random sequence $(a_1, a_2, \ldots, a_c)$ generated by the random function is used to scramble the image E_k. Then store the pseudo-random sequence in array X_k, $X_k = (a_1, a_2, \ldots, a_c)$ and establish the corresponding relation between the array subscript and the pixel position. The order of the elements in the array X_k, i.e., the position of each pixel can be reordered, so the purpose of scrambling the difference image E_k can be achieved

Step 3: Construct the histogram H_{RK} of each scrambled difference image E_{Rk}, shift and prepare empty bins in each histogram H_{RK} according to the embedding level L. Figure 3 describes a histogram shifting procedure when embedding level is 2. The shifted pixel e'_R can be computed with Eq. 8:

$$e'_R = \begin{cases} e_R + L + 1 & e_R > L \\ e_R - L - 1 & e_R < -L \\ e_R & \text{otherwise} \end{cases} \quad (8)$$

Step 4: Scan each modified difference image E'_{RK}. According to the embedding level L, the secret bit string w, w∈ [0,1], is not embedded until L<0 in the range of [-L, L] and L is decreased by 1 one by one. The detailed process is as follows: Sequentially scan each pixel e'_R, and when faced with $e'_R = \pm L$, hide the secret bit and modify the pixel values; then make L = L-1, rescan e'_R; when

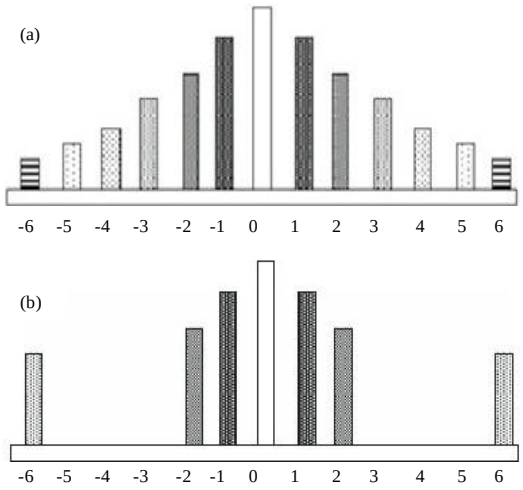

Fig. 3(a-b): Histogram shifting process (L = 2) (a) Original histogram and (b) Shifted histogram

faced with $e'_R = \pm(L-1)$, hide the data and repeat the above data embedding process until L<0. Compute the marked pixels e'_{RW} according to Eq. 9 and the marked difference images are indicated by E'_{RkW}:

$$e'_{RW} = \begin{cases} 2L + w & e'_R = L \\ -2L - w & e'_R = -L \\ e'_R & \text{otherwise} \end{cases} \quad (9)$$

Step 5: According to the corresponding relation between the array X_k subscript and the pixel position in Step 2, recover the pixels in each difference image E'_{RkW} back to their original positions to obtain the marked difference images E'_{kW}. Finally, obtain the marked sub-sampled images S_{kW} by using the reference S_{ref}

Step 6: The reference S_{ref} and the marked sub-sampled images S_{kW} are adopted for reverse sub-sampling to obtain the stego image I′. Δu, Δv, L and X_k are outputted to serve as overhead information O_{info} for decoding

Data extraction and image recovery process: After the stego image I′ and overhead information O_{info} have been obtained, Δu, Δv, L and X_k are extracted firstly, and then the embedded data bits are extracted and the cover image is recovered. The detailed procedure is listed below:

- **Input:** A stego image I′ and overhead information O_{info}, including Δu, Δv, L and X_k
- **Output:** A cover image I and secret bit w

Step 1: Extract Δu, Δv, L and X_k important information from O_{info}; obtain the marked sub-sampled image series S_k and the reference S_{ref} by using Eq. 1 and 2; and then establish the difference image series E'_{kW}, where, k = 1, 2,..., ref-1, ref+1,...., Δu×Δv

Step 2: Scramble each difference image E'_{kW} to get marked difference images E'_{RkW}. Obtain the array X_k from O_{info} and then reorder the pixel positions according to the sequence in array X_k

Step 3: The secret data extraction is a reverse embedding process. Firstly, set the embedding level L′ = 0, scan each pixel in the E'_{RkW} and retrieve all secret data bits w_0 when the embedding level is L′ = 0 by using Eq. 10. Then, make L′ = L+1, and take out all secret data bits w_1 when the embedding level is L′ = 1. Repeat the above extraction process until L′ = L:

$$w = \begin{cases} 0 & e'_{RW} = 2L' \\ 1 & e'_{RW} = 2L'+1 \\ 0 & e'_{RW} = -2L' \\ 1 & e'_{RW} = -2L'-1 \end{cases} \quad (10)$$

Step 4: Re-construct the secret bit string extracted in step 3 to obtain:

$$w = [w_L, w_{L-1},..., w_1, w_0] \quad (11)$$

After the secret data removed, the pixel values e'_R in the difference images E'_{Rk} are as follows:

$$e'_R = \begin{cases} e'_{RW} - w - L' & e'_{RW} \geq 0 \\ e'_{RW} + w + L' & e'_{RW} < 0 \end{cases} \quad (12)$$

Step 5: Shift the histogram of each E'_{RK} back to its original position. The shifted pixel e_R is:

$$e_R = \begin{cases} e'_R - (L+1) & e'_R > 2L+1 \\ e'_R + (L+1) & e'_R < -2L-1 \\ e'_R & \text{otherwise} \end{cases} \quad (13)$$

Step 6: Recover the pixels in each E'_{Rk} to the original positions to obtain the difference images E'_k. Then, obtain the sub-sampled image series S_k with the reference S_{ref}. At last, the cover image I is recovered through the inverse of the sub-sampling with the sub-sampled image series

and the reference S_{ref}.Overflow and underflow in the proposed method refers to the method in literature (Kim *et al*., 2009)

A simple example: An example of the proposed method is described in the following. One image of 6×6 pixels shown in Fig. 4a is taken as the cover image I. Let us assume that the sub-sampling factors, Δu and Δv, are also set to 2 and the embedding level L is set to 1. Perform sub-sampling from the cover image I, sub-sampled images are shown in Fig. 4b. S_1 in Fig. 4b is selected as the reference sub-sampled image and then the difference images are obtained as shown in Fig. 4c. Suppose the pseudo-random sequences (4,3,2,1), (2,1,4,3), (3,1,2,4) are applied to reorder the pixels in three difference images, respectively. The scrambled difference images are shown in Fig. 4d. Suppose the secret data is $(1101011101)_2$. Using Eq. 8-9, the shifted pixel e'_R and marked difference e'_{RW} are calculated, respectively, as shown in Fig. 4e and 4f. The same pseudo-random sequences are used again to recover the pixels back to their original locations. Figure 4g and h list the marked difference images and marked sub-sampled images of this example. Finally, the reverse sub-sampling is executed to generate the stego image I′ as shown in Fig. 4i. From that, the secret data is embedded and a stego image is obtained.

The example shown in Fig. 4 is used again to describe the proposed extraction and recovery procedure. The received stego image I′ is shown in Fig. 4i, sub-sampling factors and embedding level, are Δu = Δv = 2 and L = 1, respectively. Generate the sub-sampled marked images by performing sub-sampling from stego image, as shown in Fig. 4h and g shows the difference images among the reference S_1 and other marked sub-sampled images. Using the received pseudo-random sequences (4,3,2,1), (2,1,4,3), (3,1,2,4) reorder the pixels, respectively and results are shown in Fig. 4f. In data extraction procedure, a new variable L′ is first set to 0. Pixels in Fig. 4f are sequentially scanned. Once pixels with the value of 0 or 1 are encountered, secret data bits are retrieved with the Eq. 10, where secret data $w_0 = (01011101)_2$ is retrieved. Then, L′ is increased by 1, the secret data bits $w_1 = (11)_2$ is retrieved again. After that, the embedded secret data $(1101011101)_2$ is extracted. Remove the secret data and the results are shown in Fig. 4e. Using Eq. 13, e_R is calculated as shown in Fig. 4d. The same pseudo-random sequences are used to recover the original order of pixels in the difference images. Fig. 4c and b list the recovered difference images and sub-sampled images of this example, respectively. Finally, the inverse of sub-sampling is employed on the sub-sampled images to obtain the recovered original image I as shown in Fig. 4a.

EXPERIMENTAL RESULTS AND ANALYSIS

The goal of steganography is to evade statistical detection. In this section, the security of the proposed method under SPAM steganalyzer is analyzed by comparing with Kim's method. All the test images used in this section were obtained from RSP image database (RSP image database online (http://dud.inf.tu-dresden.de/~westfeld/rsp/rsp.html), where some literatures (Wang *et al*., 2010; Hong *et al*., 2011) also adopted this

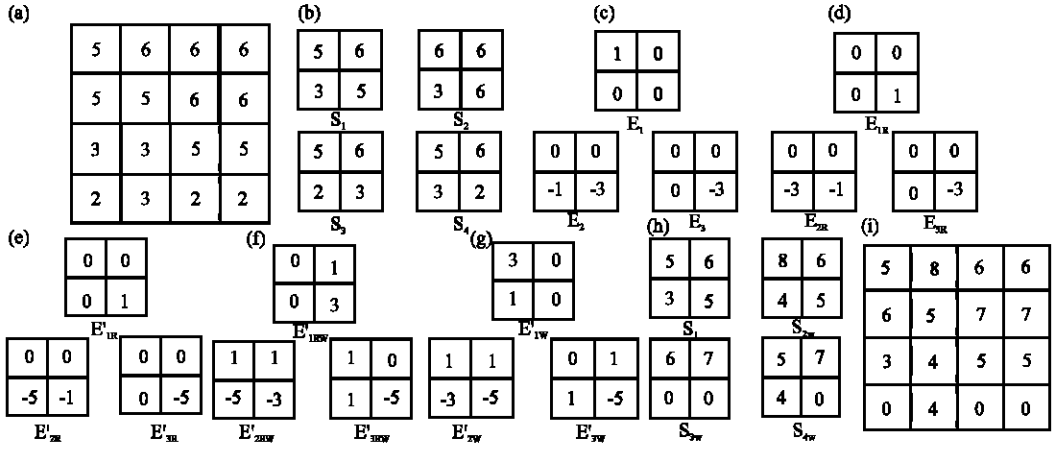

Fig. 4(a-i): An example of the proposed method (a) The cover image, (b) Sub-sampled images, (c) Difference images, (d) Scrambled difference images, (e) Shifted and scrambled difference images, (f) Marked and scrambled difference images, (g) Marked difference images, (h) Marked sub-sampled images and (i) The stego image

database for their experiments. The secret data bits to be embedded and pseudo random sequences were randomly generated using the MATLAB function. Embedding variables Δu, Δv were set to 3, the embedding levels L were chosen to be 2, 3, 5 and 7. Different amount of payload, varying from 0.01-0.05 bpp, was embedded to measure the security performance of the proposed method. A soft-margin Support Vector Machine (SVM) with Gaussian kernel was used to implement the SPAM steganalyzer. The error rate:

$$P_{Err} = \frac{1}{2}(P_{Fp} + P_{Fn}) \quad (14)$$

where, is calculated to evaluate the security of a data-hiding method against the detection of SPAM, here P_{FP} and P_{Fn} represent the false alarm rate and false denying rate, respectively. The higher the error rate of data hiding method against SPAM detection, the better its secrecy.

To evaluate the detectability of our proposed method using SPAM, we trained the SPAM steganalyzer on images obtained from RSP image database. RSP consists of 10,000 grayscale images with size 512×512 coming from cropped and resized natural images. In the experiment, 5,000 images were selected, where 2500 images were used for data hiding and other 2500 for detection instead of data hiding. SPAM obtained the features of images and then submitted them to the Support Vector Machine (SVM) for classified detection. The calculation of classification error adopted the five-fold cross-validation. In order to make the error rate the lowest, the Simulated Annealing (SA) optimization was used to find the penalization parameter g and the kernel parameter γ of SPAM.

Figure 5 shows the error rate of the proposed method against SPAM detection at various embedding capacities up to 0.5 bpp when embedding levels L = 2, L = 3, L = 5 and L = 7 were chosen.

Figure 5 shows that, whatever the embedding level is and the more the embedding capacities are, the weaker the security of our proposed method against the detection of SPAM is. Moreover, when the embedding capacity is a determinate value, the bigger the embedding level is, the more easily the embedded data will evade the SPAM detection. For example, when the embedding capacity is 0.05 bpp and embedding levels are L = 2,3,5 and 7, the error rates of the proposed method using the SPAM are 1.26%, 6.8%, 30.9% and 40.1%, respectively. This is due to the secret data embedding is achieved by using the pixel values in the range of [-L,+L]. During the data embedding, the bigger the embedding level is, the stronger the ability to hide data per pixel is. Therefore, with the same embedding capacity, fewer pixels will be used to carry data. After being randomly sorted, these

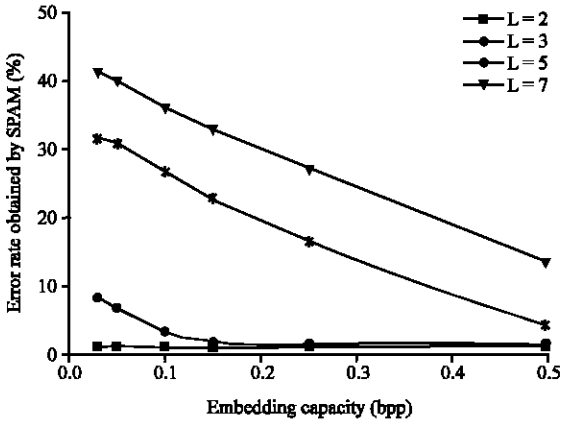

Fig. 5: Error rate obtained by SPAM for the proposed method

pixels are distributed in a relatively scattered manner. In other words, it is more likely that embedded data is subject to irregularity and is more liable to evade the SPAM detection.

Figure 6a-d show the comparison results of the security under SPAM detection for the proposed method and Kim's method at various embedding capacities up to 0.5 bpp.

As can be seen from Fig. 6, at embedding level L = 3 and above, the error rates obtained by the proposed method are significantly higher than those obtained by Kim's method, indicating that the proposed method is less detectable than Kim's method under the same payload. For example, for the embedding level at L = 7, when the embedding capacity is 0.1 bpp, the error rate of the proposed method using SPAM is 36.20%, whereas for Kim's method, only 34.42% can be obtained. That is, the gain in the error rate is 1.78% at 0.1 bpp. The undetectability is significantly higher than that of Kim's method. When the embedding level is lower at L = 2, in comparison to the error rate of Kim's method using SPAM, only 71% data of the proposed method is improved and some individual data is decreased. This is due to the reason when the embedding level is lower, the weaker the ability to hide data per pixel is. Therefore, under the same embedding capacity, more pixels will be used to carry data. After being randomly sorted, these pixels are distributed in a concentrated manner. Consequently, when the embedding level is lower, the proposed method obtains less remarkable superiority.

Tables 1 and 2 show the error rate of SPAM detection for the proposed method and Kim's method with the payloads 0.25 and 0.5 bpp. These two embedding rates were also used in (Goljan *et al*., 2006; Cancelli *et al*., 2008; Pevny *et al*., 2010). As shown in Tables 1 and 2, the error

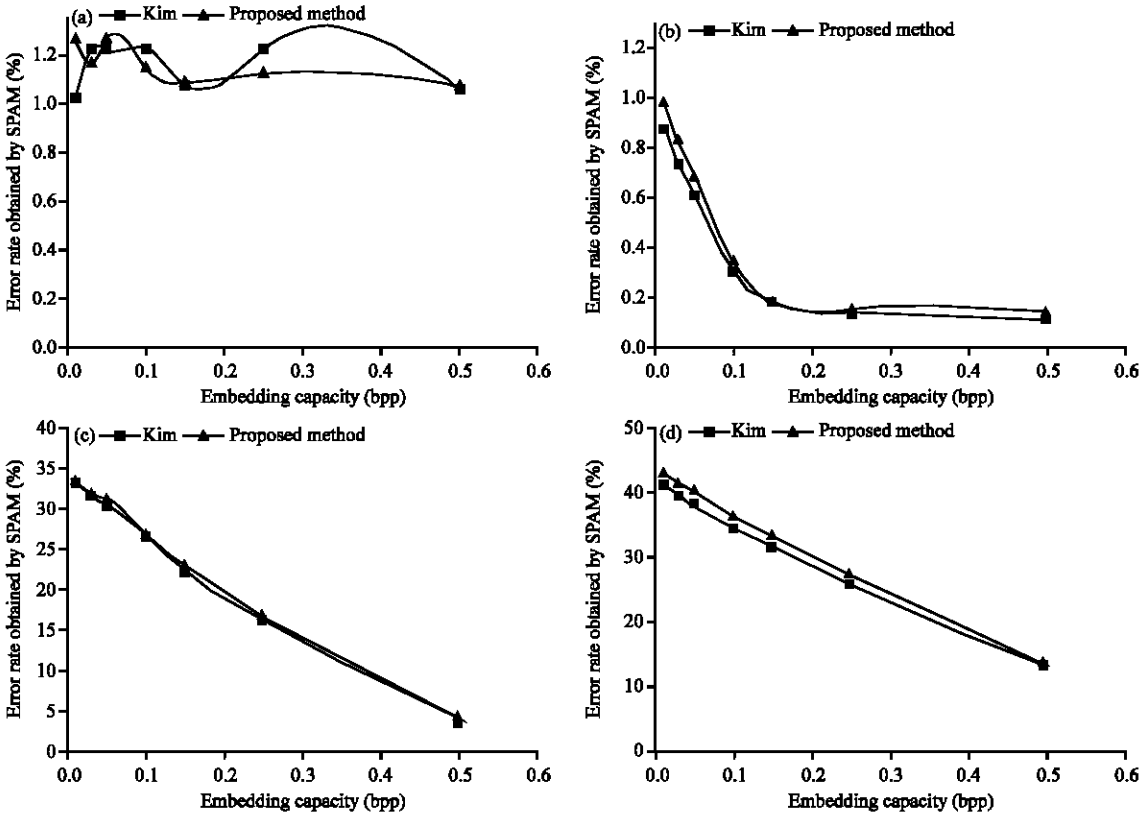

Fig. 6(a-d): Comparison of error rate obtained by SPAM for the proposed method and Kim's method (a) L = 2, (b) L = 3, (c) L = 5 and (d) L = 7

Table 1: Comparison of error rate obtained by SPAM (L = 2, 3)

	L = 2		L = 3	
Capacity (bpp)	Kim *et al.*(2009) method (%)	Proposed method (%)	Kim *et al.* (2009) method (%)	Proposed method (%)
0.25	1.22	1.26	1.36	1.52
0.5	1.06	1.07	1.12	1.40

Table 2: Comparison of error rate obtained by SPAM (L = 5, 7)

	L = 5		L = 7	
Capacity (bpp)	Kim *et al.*(2009) method (%)	Proposed method (%)	Kim *et al.* (2009) method (%)	Proposed method (%)
0.25	16.08	16.60	25.66	27.20
0.5	4.11	4.23	13.06	13.40

rates obtained by the proposed method are significantly higher than those obtained by Kim's method. For example, with the payload 0.25 bpp and embedding level L = 7, the error rate of the proposed method using SPAM is 27.20%, while error rate of Kim's method is 25.66%. The experimental results agreed with the fact that the proposed method is more secure against SPAM steganalyzer than Kim's method.

Moreover, the proposed method also provided a good visual quality. Four 512×512 cover images as seen in Fig. 7 were used to test the performance of the proposed method.

Figure 8 shows the stego-image quality of test images for the proposed method with embedding level L = 7 and payload is 0.5 bpp. Measurement for the stego-image quality is based on Peak Signal to Noise Rate (PSNR) values in dB. PSNR can be obtained by using Eq. 15:

$$PSNR = 10 \times \log_{10}\left(\frac{255^2}{MSE}\right) \quad (15)$$

where, MSE is the mean square error between the cover image and the stego image and can be computed with Eq. 16:

$$MSE = (L+1)^2 \times \left(\frac{\Delta u \cdot \Delta v - 1}{\Delta u \cdot \Delta v}\right) \quad (16)$$

As can be seen from Fig. 8, at L = 7, the proposed method achieved the PSNR values from 40.88 to 44.59 dB for all the test images and obtained a good visual quality.

Figure 9 shows that visual impacts of marked lena image obtained by different embedding levels with payload 0.5 bpp. The figure shows that the stego image quality depends on embedding level. When the

Fig. 7(a-d): Original cover images (a) Lena, (b) Boats, (c) Airplane and (d) Baboon

Fig. 8(a-d): Visual quality with L = 7 (a) Marked Lena (44.59 dB), (b) Marked boats (40.88 dB), (c) Marked airplane (44.10 dB) and (d) Marked baboon (43.26 dB)

Fig. 9(a-d): Marked Lena images obtained by the proposed method with payload 0.5bpp (a) L = 2 (49.55 dB), (b) L = 3 (47.37 dB), (c) L = 5, (44.75 dB) and (d) L = 7 (43. 26 dB)

embedding level rises high, at L = 7, the PSNR values of the proposed method are over 40 dB. As shown in Fig. 8-9, the visual quality of the marked images is satisfactory at moderate payload. That means the proposed method obtains low distortion while assuring the undetectability.

CONCLUSION

In this study, a reversible data hiding algorithm with randomly-embedded anti-SPAM detection was proposed by applying the random embedding and histogram modification technique into data hiding. By using random pixels selecting approach, the proposed method not only can evade SPAM detection successfully but also achieve low distortion. The experimental results show that the proposed method is superior to Kim's method in terms of anti-SPAM detection. The detection error rate of our proposed method can be improved by 2% to the maximum extent. However, when the embedding level is decreased, the proposed algorithm displays a less obvious advantage. The next study focus is to explore how to improve the error rate of the proposed method against SPAM detection when the embedding level is lower.

ACKNOWLEDGMENTS

This study is financially supported by the National Science Foundation (Grant No. 61103246), the National Science Council of the republic of China (Grant No. 100-221-E-025-014-MY2) and the S and T JK projects of Fujian Provincial Department of Education (grant No. JK2012044).

REFERENCES

Cancelli, G., M. Barni, G. Doerr and I.J. Cox, 2008. A comparative study of ±1 steganalyzers. Proceedings of the IEEE International Workshop on Multimedia Signal Processing, January 27-31, 2008, Queensland, Australia, pp: 791-796.

Goljan, M., J. Fridrich and T. Holotyak, 2006. New blind steganalysis and its implications. Proceedings of the SPIE 6072, Security, Steganography and Watermarking of Multimedia Contents VIII, January16-19, 2006, San Jose, CA, pp: 1-3.

Hong, W. and T.S. Chen, 2010. A local variance-controlled reversible data hiding method using prediction and histogram-shifting. J. Syst. Software, 83: 2653-2663.

Hong, W. and T.S. Chen, 2011. Reversible data embedding for high quality images using interpolation and reference pixel distribution mechanism. J. Vis. Communi. Image Represent., 22: 131-140.

Hong, W., J. Chen, T.S. Chen and C.W. Shiu, 2011. Steganography for block truncation coding compressed images using hybrid embedding scheme. Int. J. Innov. Comput., Inf. Control, 7: 733-743.

Hong, W., T.S. Chen, K.Y. Lin and W.C. Chiang, 2010. A modified histogram shifting based reversible data hiding scheme for high quality images. Inform. Technol. J., 9: 179-183.

Kim, K.S., M.J. Lee, H.Y. Lee and H.Y. Lee, 2009. Reversible data hiding exploiting spatial correlation between sub-sampled images. Pattern Recognit., 42: 3083-3096.

Li, Y.C., C.M. Yeh and C.C. Chang, 2010. Data hiding based on the similarity between neighboring pixels with reversibility. Digital Signal Process., 20: 1116-1128.

Luo, H., F.X. Yu, H. Chen, Z.L. Huang, H. Li and P.H. Wang, 2011. Reversible data hiding based on block median preservation. Inform. Sci., 181: 308-328.

Ni, Z., Y.Q. Shi, N. Ansari and W. Su, 2006. Reversible data hiding. IEEE Trans. Circ. Syst. Video Technol., 16: 354-362.

Pan, C.L., W. Hong, T.S. Chen, J. Chen and C.W. Shiu, 2011. Multilevel reversible data hiding using modification of prediction errors. Int. J. Innov. Comput., Inf. Control, 7: 5107-5118.

Pevny, T., P. Bas and J. Fridrich, 2010. Steganalysis by subtractive pixel adjacency matrix. IEEE Trans. Inform. Forensics Security, 5: 215-224.

Tian, J., 2003. Reversible data embedding using a difference expansion. IEEE Trans. Circ. Syst. Video Technol., 13: 890-896.

Wang, J., Y. Sun, H. Xu, K. Chen, H.J. Kim and S.H. Joo, 2010. An improved section-wise exploiting modification direction method. Signal Process., 90: 2954-2964.

Zhao, Z. and H. Luo, 2012. Reversible data hiding based on Hilbert curve scan and histogram modification. Inform. Technol. J., 11: 209-216.