Histogram Modification Data Hiding Using Chaotic Sequence

Xiaobo Li and Quan Zhou

Abstract In order to solve the security problem of data hiding with obvious detectable traces to attacker, a histogram modification data hiding scheme using chaotic sequence is presented. This algorithm modifies a cover image histogram by using a chaotic sequence to conceal the embedding process, and then secret information can be embedded into the modified image with encrypted trace. Experimental results show that the algorithm can achieve sound invisibility and large embedding capacity while guaranteeing high security. With the proposed scheme, the hiding trace of secret data is concealed. Even though the attacker detects the existence of hidden message under stego-image, the secret message cannot be extracted without private keys.

Keywords Data hiding • Histogram modification • Chaotic sequence • Embedding capacity

1 Introduction

With the development of internet and demand of users, image, video and multimedia data transmission in the internet increased rapidly over the last few years. Therefore, data security becomes more and more important. Data hiding [1] is a data security technique to undetectably insert secret message into a cover media to create a stego medium such that the existence of hidden information will not be detected by attacker. At the receive end, the hidden information can be extracted correctly by legal users. i.e., it provides a safer and securer data communication manner.

X. Li (✉) • Q. Zhou
Key Laboratory of Space Microwave Technology, China Academy of Space Technology, Xi’an, China
e-mail: lxb619@126.com

W. Wang (ed.), *Mechatronics and Automatic Control Systems*, Lecture Notes in Electrical Engineering 237, DOI 10.1007/978-3-319-01273-5_98,

A number of data hiding methods were proposed. There are three general types: spatial domain methods [2], frequency domain methods [3] and compression domain methods [4]. Spatial domain data hiding manner mixes secret data into the distributed pixels directly. A commonly used method, called the least significant bit (LSB) [5] method, is a simple spatial domain data hiding method by replacing the least significant bit of cover image pixels to embed secret bits. For the frequency domain manner, firstly, the cover image pixels must be transformed into frequency coefficients by using a frequency transform method such as the discrete cosine transformation (DCT) [6] and discrete wavelet transformation (DWT) [7]. Later, the secret data are embedded by modifying the relative coefficients in the frequency-form image. Finally, stego image can be obtained by utilizing corresponding inverse transform. The compression domain method means that the secret message are embedded into the compression codes, such as block truncation coding (BTC)-based [8] scheme and side match vector quantization (SMVQ)-based [9] scheme, and so on.

The desires of good data hiding schemes are high security and low image distortion as well as large payload. However, despite claiming good imperceptibility in previous schemes mentioned above, it leaves obvious detectable traces to attacker inevitably. In other words, once the attacker realizes there is a hidden information under stego-image, the secret data would be extracted possibly. To further reinforce the security of data hiding, we develop a novel histogram modification data hiding scheme by using chaotic sequence to control the embedding procedure. With this scheme, the hiding trace of secret message is concealed. Even though the attacker detects the existence of hidden message under stego image, the secret message cannot be extracted without private keys. Furthermore, the proposed scheme has nice image quality and high payload.

The rest of this paper is organized as follows. The proposed data hiding method is presented in Sect. 2. In Sect. 3, the experimental results of our method are demonstrated. Finally, the conclusions of paper are presented in Sect. 4.

2 Proposed Scheme

In this section, the proposed data hiding scheme will be presented. The proposed algorithm has the merits of high security and low image distortion as well as large embedding capacity. To enhance the security, we utilize chaotic sequence to conceal the histogram modification trace. Figure 1 shows an example of the embedding process. Figure 1a shows an image block of size 4 × 4 and histogram of the image block. Consider a binary chaotic sequence as “0110001111010110”. There is a one-to-one relationship between pixels in image block and bits of chaotic sequence. Before embedding, the image block is scanned by raster-scan order. Once an even pixel value is encountered, if the corresponding bit of chaotic sequence is “1” the pixel is added by “1”, else it is kept intact. On the contrary, once an odd pixel value is encountered, if the corresponding bit of chaotic sequence is “0” the

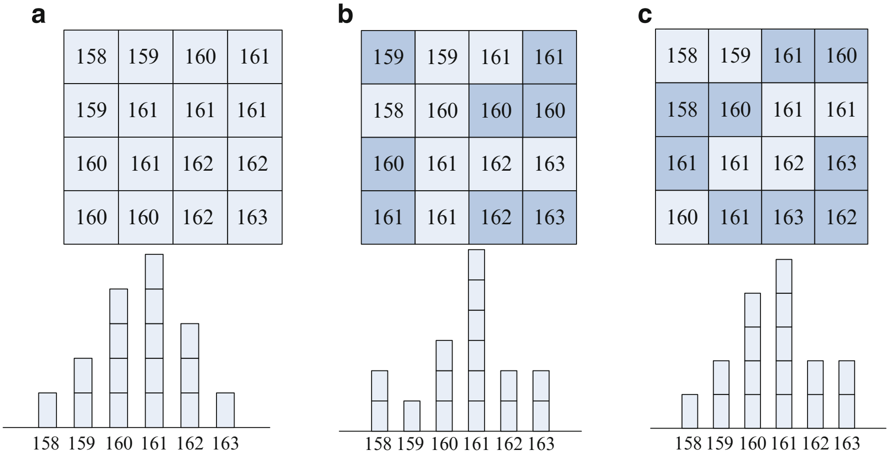


Fig. 1 An example of the embedding process : (**a**) original image block and histogram;(**b**) modified image block and histogram; (**c**) embedded image block and histogram

pixel is decremented by "1", else it is kept intact. The result of modifying operation by using chaotic sequence as shown in Fig. 1b. Suppose that the payload data is the stream of "100100111001011". To embed the stream of data, the modified image block is scanned in the same scan order once again. Whenever the same chaotic sequence '0' is encountered, we sequentially check the bit of payload data. If the payload data bit is "1" the pixel is added by "1", else it is kept intact. When the chaotic sequence "1" is encountered, if the corresponding payload data bit is "1" the pixel is decremented by "1", otherwise it is not changed. As a result, the embedded image block is obtained as shown in Fig. 1c. Data extraction is actually the reverse of the embedding process. Note that the number of payload data bits that can be hidden into an image and the number of pixels associated with the image are equal. The detail algorithm is shown as below.

2.1 Embedding Process

Assume that the cover image X of size $M \times N$ is an 8-bit grayscale image. Denote the pixel value as $X(i,j)$, where (i,j) indicates the location in the original image. We select logistic map [10] to generate chaotic sequence, as shown in formula (1):

$$l_{n+1} = \alpha \cdot l_n(1 - l_n) \tag{1}$$

Where α is bifurcation parameter, and $l_n \in (0,1)$, $n = 0,1,2\cdots$. When $3.5699456\cdots < \mu \leq 4$, the logistic map is in chaotic state. In this paper, the parameter α and initial value l_0 are used as private keys.

Step1: Select parameter α and initialize value l_0, chaotic sequences S_k can be obtained by

$$S_k = \begin{cases} 0 & if \;\; 0 < l_n < 0.5 \\ 1 & if \;\; 0.5 \le l_n < 1 \end{cases} \tag{2}$$

Where $k = 1, 2, \ldots, M \times N$, $n = 0, 1, \ldots, (M \times N) - 1$.

Step2: Scan the whole image in a given order, modify the pixel value $X(i,j)$ by

$$Y(i,j) = \begin{cases} X(i,j) & if \;\; S_k = 0 \& R_{i,j} = 0 \\ X(i,j) - 1 & if \;\; S_k = 0 \& R_{i,j} = 1 \\ X(i,j) & if \;\; S_k = 1 \& R_{i,j} = 1 \\ X(i,j) + 1 & if \;\; S_k = 1 \& R_{i,j} = 0 \end{cases} \tag{3}$$

Where $R_{i,j} = rem(X(i,j), 2)$ is the remainder of pixel value $X(i,j)$ and integer 2, $Y(i,j)$ is modified value of pixel $X(i,j)$, $1 \le i \le M$, $1 \le j \le N$, and $k = 1, 2, \ldots, M \times N$.

Step3: Scan the whole image in the same order in step2, and modify $Y(i,j)$ according to the secret message B.

$$Z(i,j) = \begin{cases} Y(i,j) + B & if \;\; S_k = 0 \\ Y(i,j) - B & if \;\; S_k = 1 \end{cases} \tag{4}$$

Where S_k is the same chaotic sequences in step2, $k = 1, 2, \ldots, M \times N$. B is a binary sequence, and $B \in \{0, 1\}$.

Finally, the final stego image Z with the embedded secret information is constructed.

2.2 Extraction Process

The secret information extraction process is similar with the embedding process exception the image is stego image. The extraction process is described as follows:

Step1: Generate chaotic sequences S_k by utilizing private keys α and l_0.

Step2: Scan the whole stego image Z in the same order as during the embedding. Extract message B by

$$B = \begin{cases} 0 & if \;\; S_k = 0 \& R_{i,j} = 0 \\ 1 & if \;\; S_k = 0 \& R_{i,j} = 1 \\ 0 & if \;\; S_k = 1 \& R_{i,j} = 1 \\ 1 & if \;\; S_k = 1 \& R_{i,j} = 0 \end{cases} \tag{5}$$

Where $R_{i,j} = rem(Z(i,j), 2)$ is the remainder of pixel value $Z(i,j)$ and integer 2, $1 \leq i \leq M$, $1 \leq j \leq N$, and $k = 1, 2, \ldots, M \times N$.

Thus, the secret message hidden in the stego image are obtained.

3 Results and Discussions

We select four 8-bits grayscale images of size 512 × 512 to evaluate the performance of the proposed scheme, as shown in Fig. 2. These natural images, "Lena", "Baboon", "Boat", and "Pepper", are selected from CVG-UGR image database [11]. To evaluate the visual quality of stego image, we use the function of peak-signal-to-noise-ratio (PSNR), which is defined as in

$$PSNR(dB) = 10 \times \log_{10}\left(\frac{255^2 \times M \times N}{\sum_{i=1}^{M}\sum_{j=1}^{N}(X(i,j) - Z(i,j))^2}\right) \tag{6}$$

Where, X and Z denote the original cover image and stego image, M and N denote the width and height of the cover image, respectively.

The results of four test images for embedding are listed in Table 1, which shows the embedded capacity and PSNR of the stego image. From the Table 1 it is seen that the values of PSNR of four test images are all greater than 51 dB. The embedding capacity of four test images are 262,144 bits, i.e. 1 bpp (bits per pixel). This demonstrates that our proposed algorithm can achieve nice invisibility and large embedding capacity.

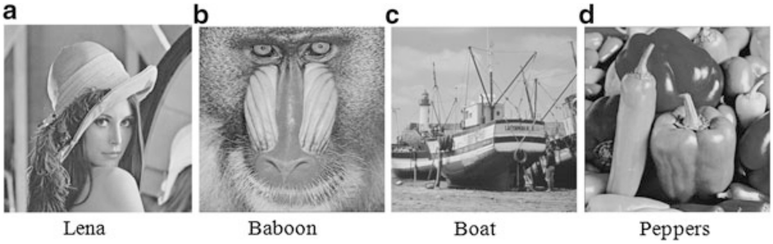


Fig. 2 Original cover images (**a**) Lena (**b**) Baboon (**c**) Boat (**d**) Peppers

Table 1 Embedding capacity and distortion for test images

Test images	Capacity (bits)	PSNR (dB)	Key
Lena	262,144	51.13	Yes
Baboon	262,144	51.13	Yes
Boat	262,144	51.14	Yes
Peppers	262,144	51.13	Yes

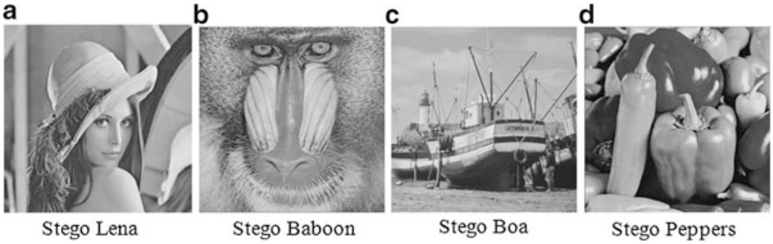


Fig. 3 The stego images (**a**) Stego Lena (**b**) Stego Baboon (**c**) Stego Boa (**d**) Stego Peppers

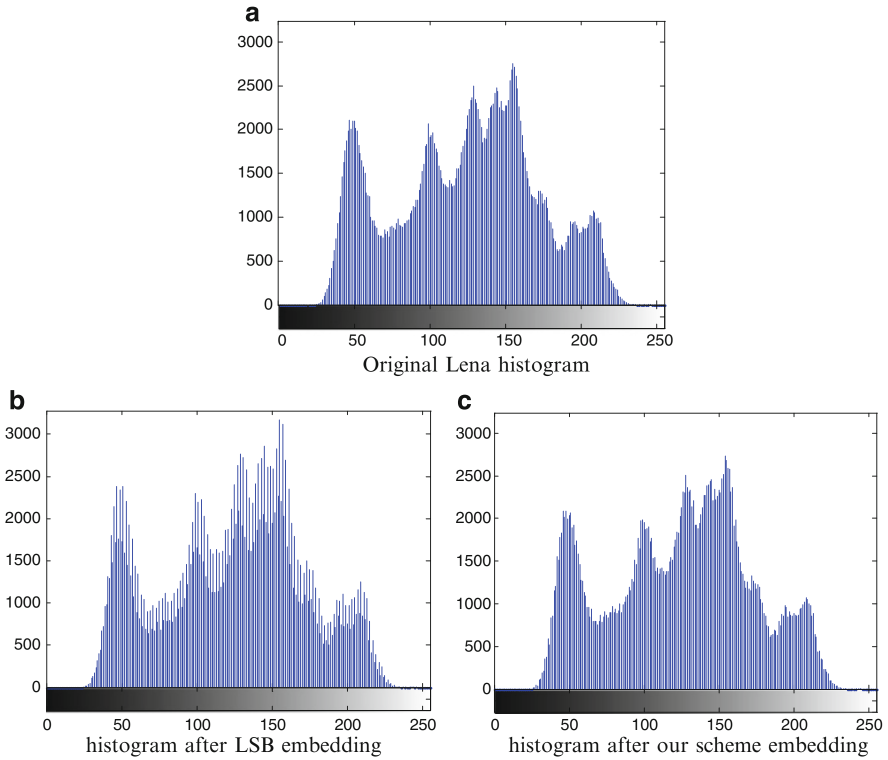
Figure 3 shows the visual quality of stego images when embedding same secret data with length of 262,144 bits. The visual differences cannot be detected by the Human Vision System (HVS) between the stego images and the corresponding original cover images. This is the most important desire for data hiding application.

Figure 4 shows the histogram distribution of Lena image after embedding data by using LSB method and our proposed method, respectively. Figure 4a is the histogram of original Lena image. Figure 4b is the histogram of stego image obtained from applying the classical LSB data hiding approach. From Fig. 4b, the histogram of stego image has changed obviously. it leaves detectable traces to attacker inevitably. In other words, once the attacker realizes there is a hidden information under stego image, the secret data would be extracted possibly by analysing the distribution of histogram of the stego image. Figure 4c is the histogram of stego image obtained from applying our proposed data hiding method. With reference to Fig. 4c we can clearly see that there is almost no change on histogram between stego image and original image. i.e., the attacker cannot detect the changes done for data hiding.

The result of security test for our proposed method is showed in Fig. 5. Figure 5a is a secret binary image with size 512 × 512. We embed this secret binary image into cover image by applying our proposed embedding process with private keys, and then extract the hidden information under stego image by applying our proposed extraction process with wrong keys. Finally, the extracted image is shown in the Fig. 5b. From the Fig. 5b it is seen that the extracted image is similar to white noise image, which demonstrates the extracted image varies sensitively with the variances of the private keys.

4 Conclusion

This paper put up a histogram modification data hiding method using chaotic sequences for increasing the security. The secret message is embedded at different histogram modification traces, which are encrypted by private keys. Even though the attacker detects the existence of hidden message under stego image, the secret message cannot be extracted without private keys. Experimental results

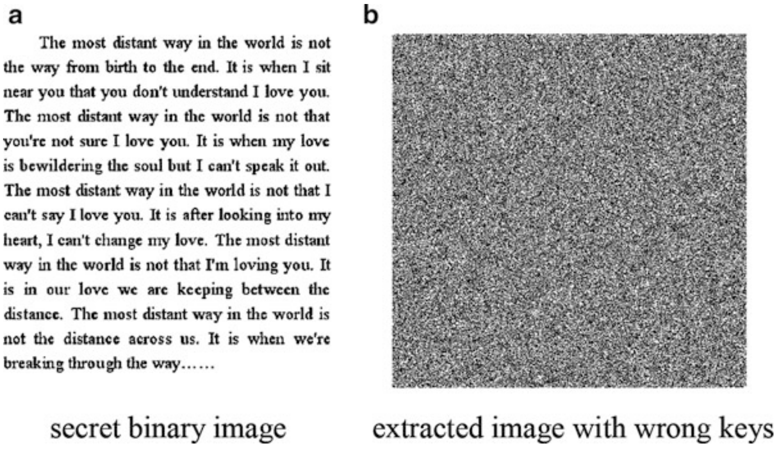


Original Lena histogram

histogram after LSB embedding

histogram after our scheme embedding

Fig. 4 The original Lena histograms and the stego image histograms with different embedding methods (**a**) original Lena histogram, (**b**) histogram after LSB embedding (**c**) histogram after our scheme embedding



The most distant way in the world is not the way from birth to the end. It is when I sit near you that you don't understand I love you. The most distant way in the world is not that you're not sure I love you. It is when my love is bewildering the soul but I can't speak it out. The most distant way in the world is not that I can't say I love you. It is after looking into my heart, I can't change my love. The most distant way in the world is not that I'm loving you. It is in our love we are keeping between the distance. The most distant way in the world is not the distance across us. It is when we're breaking through the way……

secret binary image

extracted image with wrong keys

Fig. 5 Security test (**a**) secret binary image (**b**) extracted image with wrong keys

demonstrated that the proposed data hiding scheme can provide higher security while keeping low distortion and large embedding payload. Besides, the embedding process of the proposed scheme just deals with the scanning, adding, and subtracting operations that the computation complexity of this scheme is also very small. It is expected that the proposed scheme having high security can be deployed for extensive application fields.

References

1. Petitcolas FAP, Anderson RJ, Kuhn MG (1999) Information hiding-a survey. Proc IEEE 87 (7):1062–1078
2. Wu DC, Tsai WH (2000) Spatial-domain image hiding using image differencing. IEE Proc Vis Image Signal Process 147(1):29–37
3. Bao P, Ma X (2005) Image adaptive watermarking using wavelet domain singular value decomposition. IEEE Trans Circuits Syst Video Technol 15(1):96–102
4. Chuang JC, Chang CC (2006) Using a simple and fast image compression algorithm to hide secret information. Int J Comput Appl 28(4):329–333
5. Chan CK, Cheng LM (2004) Hiding data in images by simple LSB substitution. Pattern Recog 37(3):469–474
6. Singh S, Siddiqui TJ, Singh R et al (2011) DCT-domain robust data hiding using chaotic sequence. In: International conference on multimedia, signal processing and communication technologies, IEEE Press, Aligarh, India, pp 300–303
7. Huang HY, Chang SH (2010) A lossless data hiding based on discrete Haar wavelet transform. In: IEEE 10th international conference on Computer and Information Technology (CIT), IEEE Press, Bradford, England, pp 1554–1559
8. Chang CC, Lin CY, Fan YH (2008) Lossless data hiding for color images based on block truncation coding. Pattern Recog 41(7):2347–2357
9. Lee JD, Chiou YH, Guo JM (2010) Reversible data hiding based on histogram modification of SMVQ indices. IEEE Trans Info Forensics Secur 5(4): 638–648
10. Sun Y, Wang GY (2011) An image encryption scheme based on modified logistic map. International workshop on Chaos-Fractals Theories and Applications(IWCFTA), IEEE Press, Hangzhou, China, pp 179–182
11. CVG-UGR Image Database: http://decsai.ugr.es/cvg/dbimagenes/index.php