Robust Lossless Data Hiding Using Chaotic Sequence and Statistical Quantity Histogram

Xiaobo Li(✉), Jianhua Zhang, and Quan Zhou

China Academy of Space Technology, Xi'an, People's Republic of China
lxb619@126.com

Abstract. In order to solve the security problem of robust lossless data hiding, A method to cover up detectable traces is proposed based on chaotic sequence and statistical quantity histogram. The proposed method divides a original image into non-overlapping blocks and selects a smaller block in each block by using chaotic sequence. Then calculates the statistical quantity of each selected block as a robust parameter and shifts it by appropriate thresholds. Finally, secret information can be hidden into blocks by modifying robust parameter values. With our proposed method, the embedding trace of secret information is concealed. Experimental results show that the proposed method can achieve high performances in security, robustness and visual quality.

Keywords: Robust lossless data hiding · Statistical quantity · Chaotic sequence

1 Introduction

Recently, many information hiding methods have been proposed [1–5]. However, most lossless data hiding methods are exposed to an open environment to deliver the hidden information, and thus they are not safety for practical applications [6–11]. To solve the problems, we propose a novel robust information hiding method. The proposed method divides a carrier image into a number of non-overlapping blocks and selects a smaller block of each block by utilizing chaotic sequence. Then calculates the robust parameter of each selected block called statistical quantity. Secret information are hidden into blocks by shifting the robust parameter histogram. Without private keys, the secret information can't be extracted, even if the third party detects the existence hidden information under carrier image. Experimental results show that the proposed method has higher visual quality and better robustness than that of other robust lossless data hiding methods being mentioned in the paper.

The rest of this paper is organized as follows, the proposed method is presented in Sect. 2. Experimental results are discussed in Sect. 3. Finally, the conclusions of paper are given in Sect. 4.

B. Li et al. (Eds.): IoTaaS 2018, LNICST 271, pp. 461–468, 2019.
https://doi.org/10.1007/978-3-030-14657-3_47

2 Proposed Method

2.1 Robust Parameter

First of all, an 8-bit original image with size $M \times N$, denoted by I, is divided into a number of un-overlapping blocks each of size $m \times n$. And divide the $m \times n$ block into un-overlapping smaller blocks with size $h \times w$ consecutively. Then, we can establish a one-to-one mapping between $m \times n$ blocks and random integers of chaotic sequence. Based on the above results, we introduce a $h \times w$ matrix E, the matrix E is shown below:

$$E(i,j) = \begin{cases} -1 & if \quad \mathrm{mod}(i,2) \neq \mathrm{mod}(j,2) \\ 1 & if \quad \mathrm{mod}(i,2) = \mathrm{mod}(j,2) \end{cases}$$

Where $i \in [1,h]$, $j \in [1,w]$ and mod(i, 2) or mod(j, 2) is a mod-2 function.
The robust parameter of selected bock is given by

$$\alpha^{(Z)} = \sum_{i}^{h} \sum_{j}^{w} (C_Z(i,j) \times E(i,j))$$

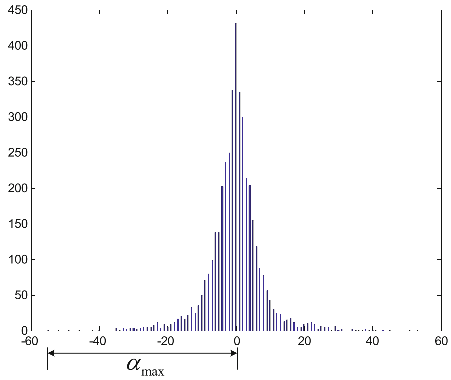

Fig. 1. The distribution of α of image "Lena" with $m \times n = 8 \times 8$ and $h \times w = 2 \times 2$

Where $C_z = \{C_z(i,j)|C_z(i,j) \in \{0,1,\ldots,255\},\ i = 1,2,\ldots,h,\ j = 1,2,\ldots,w\}$, and the subscript ($Z$) means the Z-th selected block with size $h \times w$, and $\alpha^{(Z)}$ means the robust parameter value of the Z-th selected block. Because small changes to the blocks caused by attacks such as JPEG2000 compression will not cause the robust parameter α to change much, and robustness is achieved. Thus, the statistical quantity of α can be used to hide information. The distribution of α is shown in Fig. 1, where the x-axis being the value of α, while the y-axis is the number of α. Supposing that α_{max} is the maximum absolute value between the values of α. From the Fig. 1 it is seen that the α values are close to zero, that is because pixels in a local area have a good relativity. This distribution of α can be used to produce extra space for embedding information.

2.2 Information Hiding Process

In this part, we will use the statistical quantity histogram formed by α as the hiding carrier for information embedding. First, we introduce four thresholds, denoted by T_1, T_2, R_1, and R_2, both of which are respectively positive integers. We let $0 < T_1 \leq \alpha_{max}$, $T_2 = \alpha_{max} - T_1$, and R_1, R_2 are used to separate the different zones as robust thresholds.

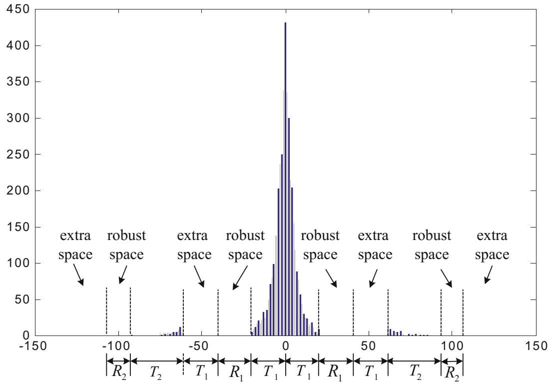

Fig. 2. Distribution of α after shifting

Next, for generating the robust space, we shift statistical quantity by modifying the pixels of selected blocks C_Z as follows:

$$D_Z(i,j) = \begin{cases} C_Z(i,j) + \beta_1 & if\ \alpha > T_1 \& E(i,j) = 1 \\ C_Z(i,j) - \beta_1 & if\ \alpha > T_1 \& E(i,j) \neq 1 \\ C_Z(i,j) - \beta_1 & if\ \alpha < -T_1 \& E(i,j) = 1 \\ C_Z(i,j) + \beta_1 & if\ \alpha < -T_1 \& E(i,j) \neq 1 \\ C_Z(i,j) & otherwise \end{cases}$$

Where $i \in [1, h]$, $j \in [1, w]$, $\beta_1 = \lceil (T_1 + R_1)/(h \times w) \rceil$.

From the Fig. 2, the resulting distribution of shifted α is shown, and it is seen that four extra spaces are obtained for information embedding, i.e., $(-\infty, -2T_1 - T_2 - R_1 - R_2]$, $[-2T_1 - R_1, -T_1 - R_1]$, $[T_1 + R_1, 2T_1 + R_1]$ and $[2T_1 + T_2 + R_1 + R_2, \infty)$. Besides, the range of $(-2T_1 - T_2 - R_1 - R_2, -2T_1 - T_2 - R_1)$, $(-T_1 - R_1, -T_1)$, $(T_1, T_1 + R_1)$ and $(2T_1 + T_2 + R_1, 2T_1 + T_2 + R_1 + R_2)$ are four robust spaces. Then, we scan each block and check up the corresponding α, the secret bits can be hidden into these blocks. The hiding process is as follows:

(1) While $-T_1 \leq \alpha \leq T_1$, the hiding regulation is given by
If $B = 0$, let $G_Z(i,j) = D_Z(i,j)$.
If $B = 1$, let

$$G_Z(i,j) = \begin{cases} D_Z(i,j) + \beta_1 & \textit{if } 0 \le \alpha \le T_1 \& E(i,j) = 1 \\ D_Z(i,j) - \beta_1 & \textit{if } 0 \le \alpha \le T_1 \& E(i,j) \ne 1 \\ D_Z(i,j) - \beta_1 & \textit{if } -T_1 \le \alpha < 0 \& E(i,j) = 1 \\ D_Z(i,j) + \beta_1 & \textit{if } -T_1 \le \alpha < 0 \& E(i,j) \ne 1 \\ D_Z(i,j) & \textit{otherwise} \end{cases}$$

(2) While $\alpha > T_1$ or $\alpha < -T_1$, the hiding regulation is given by
If $B = 1$, let $G_Z(i,j) = D_Z(i,j)$.
If $B = 0$, let

$$G_Z(i,j) = \begin{cases} D_Z(i,j) + \beta_2 & \textit{if } \alpha > T_1 \& E(i,j) = 1 \\ D_Z(i,j) - \beta_2 & \textit{if } \alpha > T_1 \& E(i,j) \ne 1 \\ D_Z(i,j) - \beta_2 & \textit{if } \alpha < -T_1 \& E(i,j) = 1 \\ D_Z(i,j) + \beta_2 & \textit{if } \alpha < -T_1 \& E(i,j) \ne 1 \\ D_Z(i,j) & \textit{otherwise} \end{cases}$$

Where $i \in [1,h]$, $j \in [1,w]$, $\beta_2 = \lceil (T_2 + R_2)/(h \times w) \rceil$. When bits are hid into the image blocks, the pixel values of blocks need to be modified by β_1 or β_2. The values of β_1 and β_2 is known as hiding level.

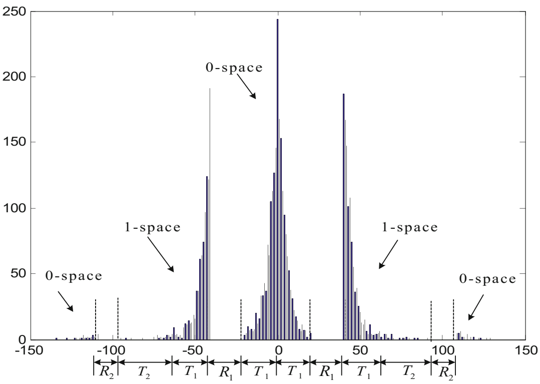

Fig. 3. Distribution of α after hiding data

The distribution of α after hiding data is shown in Fig. 3. When 0s are embedded, the ranges of α are kept within $[-T_1, T_1]$, $[-\infty, -(R_2 + T_2 + R_1 + 2T_1)]$ and $[2T_1 + R_1 + T_2 + R_2, \infty]$, and these ranges are called the 0-space; When 1 s are embedded, the ranges of α are kept within $[T_1 + R_1, 2T_1 + R_1 + T_2]$ and $[-(2T_1 + R_1 + T_2), -(T_1 + R_1)]$, these ranges are called the 1-space. Figure 3 shows the 0-space and the 1-space are separated by a distance R_1 or R_2, attacks applied to the stego-image will not cause the 0-space and the 1-space to overlap, i.e., thus, the hidden information are robust to attacks, the hidden bits can be extracted correctly. Clearly, the robust capability of hidden bits are corresponding to distance R_1 or R_2. That is, the larger the distance R_1 or R_2 is, the more robust hidden bits can be extracted correctly. Besides, the embedding capacity of our method is $Cap = \lfloor M/m \rfloor \times \lfloor N/n \rfloor$ bits.

2.3 Information Extraction

The information extraction of stego-image has two cases, one is stego-image remains intact in a lossless environment, the extracting procedure is the same as the hiding procedure exception the processing image is stego-image. For another case, once the stego-image has been attacked, for instance, JPEG2000 compression, the distribution of statistical quantity is changed and many α step into the wrong zone, as shown in Fig. 4. With the different, in addition to record the amounts of 0s and 1s in the hidden bits, denoted by N0 and N1, we must record the numbers of 0s embedded in the range $[-T_1, T_1]$ of statistical quantity α, denoted by N2. As shown in Fig. 4, we can obtain a bits_x such that the number of α in the range of [−x_bits, x_bits] is equal to N2, a bits_y such that the number of α in the range of [−y_bits, y_bits] is equal to (N2 + N1), and a bits_z such that the number of α in the range of [−z_bits, z_bits] is equal to (N0 + N1), respectively. Thus, the hidden bits can be extracted by

$$B = \begin{cases} 0 & \begin{array}{l} if\ -x\_bits \le \alpha \le x\_bits \ \ or \\ \alpha > y\_bits \ \ or \\ \alpha < -y\_bits \end{array} \\ 1 & \begin{array}{l} if\ -y\_bits \le \alpha < -x\_bits \ \ or \\ x\_bits < \alpha \le y\_bits \end{array} \end{cases}$$

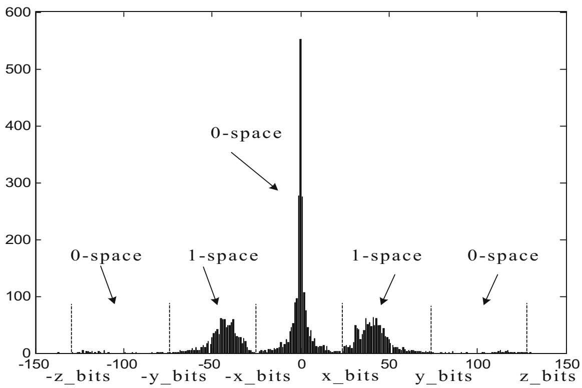

Fig. 4. Distribution of α of a stego-image has been attacked

3 Experiments

In order to verify the effectiveness of the proposed method in this paper, we test the proposed algorithm on six 8-bits grayscale images with 512 × 512 resolution, “Lena”, “Baboon”, “Boat”, “Airplane”, “Pepper”, and “GoldHill” [12]. The secret bits used in our tests were generated by a random number generator. Six stego-images were compressed by JPEG2000 under the different image compression ratio in robustness testing. Robustness against compression was measured by bpp (the survival of bit rate) at BER (a bit error rate <1%). The survival of bit rate is used to adjust image quality under the different image compression ratio, and BER means the percentage of errors bits in total hidden bits. As a general rule, the lower is the bpp and BER, the better is the robustness. We use the general visual evaluation function of PSNR to evaluate the image quality.

Table 1. The test results of the proposed method with block size 8 × 8

Images (512 × 512)	PSNR (dB)	Capacity (bits)	Threshold (T_1, T_2)	Threshold (R_1, R_2)	EL* (β_1, β_2)	Rb* (bpp)	Key (k, x_0)	BER (%)	BERW* (%)
Lena	42.29	4096	(20, 35)	(20, 5)	(10, 10)	0.62	(5, 0.5)	0.85	50.9
Airplane	42.20	4096	(20, 37)	(20, 3)	(10, 10)	0.67	(6, 0.5)	0.85	48.6
Boat	41.70	4096	(20, 32)	(20, 8)	(10, 10)	0.94	(4, 0.6)	0.88	50.2
GoldHill	42.12	4096	(20, 36)	(20, 4)	(10, 10)	0.94	(4, 0.6)	0.93	49.1
Peppers	39.79	4096	(20, 36)	(20, 4)	(10, 10)	1.23	(8, 0.3)	0.66	49.3
Baboon	32.56	4096	(60, 58)	(60, 2)	(30, 15)	1.60	(3, 0.2)	0.24	49.2

*EL = Embedding level; Rb = Robustness; BERW = BER with wrong keys.

First, the test images was divided into blocks of size 8 × 8, and select blocks with size 2 × 2 in each 8 × 8 block by using chaotic sequence with the parameter k and initial value x_0 as private keys. Then calculate statistical quantity value α of each selected block, and choose appropriate threshold T_1, T_2, R_1, and R_2 to shift the statistical quantity histogram. Finally, 4096 secret bits can be hid into six test images. Then, all the stego-images under the different image compression ratio were compressed by JPEG2000 after embedding secret bits. In the end, the hidden bits were extracted from the attacked stego-images, and the test results are shown in Table 1. In the Table 1, it is seen that the PSNR range of stego-images show that from 42.29 dB to 32.56 dB, and robustness from 0.62 to 1.60 bpp when BER < 1%. To show the security of our method, the embedded bits were extracted from the stego-images with wrong keys, Table 1 shows that the BER values are all near to 50%. This implies that hidden data extracted under stego image are completely wrong. The distribution of α after embedding data with wrong keys is show in Fig. 5. In contrast with Fig. 3, our method cover up the bits embedding trace. In general, high security and robustness while keeping the distortion low is proved by the experiment result of our method.

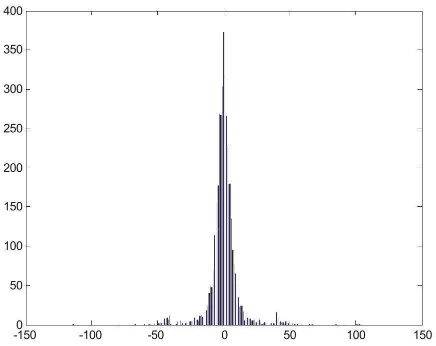

Fig. 5. The distribution of α after embedding data with wrong keys

Finally, comparing the results of our method to the Zeng et al.'s method [8], we list some experimental results of this methods on six test images. In this experiments, we used a block size of 8 × 8. A performance comparison results is shown in Table 2. Experimental results show that our proposed method can achieve higher performances in visual quality of stego-image, robustness, and data embedding capacity than other methods.

Table 2. Performance comparison results

Images (512 × 512)	Zeng's scheme				Our proposed scheme			
	PSNR (dB)	Capacity (bits)	Rb* (bpp)	BER (%)	PSNR (dB)	Capacity (bits)	Rb* (bpp)	BER (%)
Lena	38.60	4096	1.04	0.69	42.29	4096	0.62	0.85
Airplane	38.60	4096	1.05	0.80	42.20	4096	0.67	0.85
Boat	38.59	4096	1.56	0.77	41.70	4096	0.94	0.88
GoldHill	38.58	4096	1.72	0.90	42.12	4096	0.94	0.93
Peppers	37.26	4096	0.81	0.74	39.79	4096	1.23	0.66
Baboon	31.87	4096	1.70	0.94	32.56	4096	1.60	0.24

*Rb = Robustness.

4 Conclusion

The proposed method embeds secret information into carrier image by modifying the statistical quantity parameters of blocks selected by chaotic sequence. Without private keys, the secret information can't be extracted, even if the third party detects the existence hidden information under carrier image. Experimental results demonstrate that our method provides a security approach to embed information. Performance comparisons with other methods are provided to show that the proposed method has obtained an high performances in visual quality of images, information embedding capacity and robustness. It is expected that the proposed method can be applied in information safety fields.

References

1. Dragoi, I.C., Coltuc, D.: Reversible data hiding in encrypted images based on reserving room after encryption and multiple predictors. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, Calgary, AB, Canada, pp. 2102–2105 (2018)
2. Lee, C.-F., Shen, J.J., Lai, Y.H.: Data hiding using multi-pixel difference expansion. In: 2018 3rd International Conference on Computer and Communication Systems, ICCCS, Nagoya, Japan, pp. 56–60 (2018)
3. Wang, W., Ye, J., Wang, T., Wang, W.: Reversible data hiding scheme based on significant-bit-difference expansion. IET Image Process. **11**, 1002–1014 (2017)
4. Yi, S., Zhou, Y.: Adaptive code embedding for reversible data hiding in encrypted images. In: 2017 IEEE International Conference on Image Processing, ICIP, Beijing, China, pp. 4322–4326 (2017)
5. Shi, Y.-Q., Li, X., Zhang, X., Wu, H.-T., Ma, B.: Reversible data hiding: advances in the past two decades. IEEE Access **4**, 3210–3237 (2016)
6. Singh, A., Dutta, M.K.: Lossless and robust digital watermarking scheme for retinal images. In: 2018 4th International Conference on Computational Intelligence & Communication Technology, CICT, Ghaziabad, India, pp. 1–5 (2018)
7. Choi, K.-C., Pun, C.-M.: Difference expansion based robust reversible watermarking with region filtering. In: 2016 13th International Conference on Computer Graphics, Imaging and Visualization, CGiV, Beni Mellal, Morocco, pp. 278–282 (2016)
8. Zeng, X.-T., Ping, L.-D., Pan, X.-Z.: A lossless robust data hiding scheme. Pattern Recogn. **43**, 1656–1667 (2010)
9. Zeng, X.-T., Pan, X.-Z., Ping, L.-D., et al.: Robust lossless data hiding scheme. J. Zhejiang Univ.-SCIENCE C (Comput. Electron.) **11**(2), 101–110 (2010)
10. Yang, Q.T., Gao, T.G., Fan, L.: Lossless robust data hiding scheme based on histogram shifting. In: Hu, W. (ed.) Electronics and Signal Processing. LNEE, vol. 97, pp. 937–944. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21697-8_120
11. An, L., Gao, X., Yuan, Y.: Robust lossless data hiding using clustering and statistical quantity histogram. Neurocomputing **77**, 1–11 (2012)
12. CVG-UGR Image Database. http://decsai.ugr.es/cvg/dbimagenes/index.php