Encryption and Compression Scheme of Color Image Based on a Hyperchaotic System

http://dx.doi.org/10.3991/ijoe.v9iS6.2858

PENG Jing-yu[1], GONG Sheng-rong[2]
[1]College of Applied Technique, Soochow University, Suzhou, China
[2] College of Computer Science and Technology, Soochow University, Suzhou, China)

***Abstract*—In order to encrypt and compress a color image simultaneously, an encryption and compression algorithm for color image was proposed based on a hyperchaotic system. A hyperchaotic system was utilized to make a pure white image into a uniform distribution and colorful palette. In compression process, indexing technology was utilized to map the 3D data of color image to two-dimensional subscript of palette. The compressed image would continue to be encrypted by loop feedback in physical space and time space. Simulation experiments show that the algorithm is realized to encrypt and compress a color image simultaneously. It has large key space, high security, and big compression ratio as well as great application value in secure communication of color image.**

***Index Terms*—Secure Communication, Image Encryption, Index Technology, Compression Coding**

I. INTRODUCTION

Image encryption is widely utilized in finance, security, military and other fields. The amount of image data becomes larger and larger from binary image to gray image to color image. A true color image consists of pixels in image matrix. Usually per pixel is represented by red (R), green (G), and blue (B). The amount of data of an m×n true color image is m×n×3×8 bit, 24 times binary image of the same size if 8 bits are utilized to represent gray level of each color. The large amount of color image data limits the massage transmission and the storage speed. Therefore, both reliability of encryption scheme and compressibility of data should be considered in the encryption of color image. Most current encryption schemes [1-3] didn't consider compressibility of image data after encryption. In fact, many compressibility scheme of color image [4-6] cannot be utilized in encrypted images as order of arrangement as well as pixel value will change, and the correlation among pixels is disorganized. Normally the image should be encrypted after compressed [7]. In recent years, there have been some studies on the compression for encrypted images [8-9]. A resolution-progressive compression scheme was proposed in literature [8] to compress encrypted images. In literature [9], a lossy compression scheme was proposed to utilize pseudo random permutation to encrypt original image. By removing coarse coefficients in orthogonal transformation and reconstructing valid information from detail coefficients, the encrypted data can be compressed. Encryption and compression in these processes are chronological. So it is a hotspot to study on compressing and encrypting color images simultaneously [10-12].

An encrypted compression scheme for color image is proposed to compress color images while encrypting it. It is different from former encrypted schemes in that chaotic signal takes effect in a pure white image instead of a secret image. Instead of changing pixel position or pixel value, the encryption of secret image maps each pixel to a subscript in white image. Simulation experiments analyze the compression ratio of the algorithm and the compression quality. And also the statistical characteristics, key space and sensitivity of encrypted images are tested.

II. ENCRYPTED COMPRESSION SCHEME

The encrypted compression scheme is shown as Figure1. First, the hyperchaotic sequences after numerical transformation are taken effect in a white image to make a palette with rich colors and uniform distributed colors. Then the Euclidean distance is calculated and each pixel in secret image and the color in palette are compared to find out the subscript corresponding to the nearest color in palette. Finally the subscript is coded as binary data stream, and the data stream after compression and encryption can be obtained by circularly encrypting data stream.

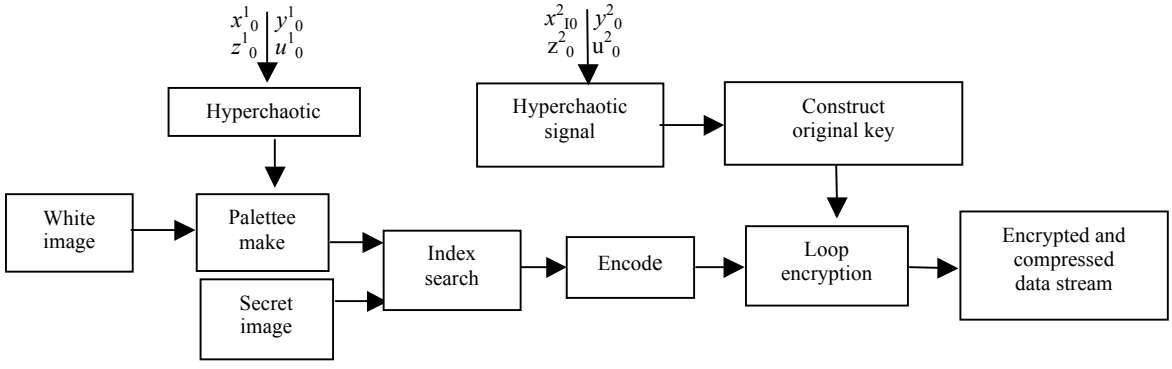

Figure 1. Process of compress and encryption

A. Making palette

1) Hyperchaotic signal

In recent years, hyperchaotic signal is widely used in encryption scheme [13-14] as its high sensitivity for initial conditions and pseudo randomness. However, its key space is small, complexity of sequences is not high, and the cryptosystem is not secure. So Changci Wen etc. proposed composed chaotic system [15], Jing Wang, Congxu Zhu etc. proposed hyperchaotic system [16-17] to increase key space and improve the safety of cryptosystem.

The hyperchaotic system shown in equation (1) is utilized to make palette. Literature [18] has proved the system is hyperchaotic.

$$\begin{cases} \dot{x} = a(y - x) \\ \dot{y} = cx - xz + u \\ \dot{z} = -bz + xy \\ \dot{u} = -dx \end{cases} \quad (1)$$

In equation (1), when a=35, b=3, c=35, d=8, Lyapunov index is k1=012788, k2=011470, k3=0, k4=-381429. There are two positive Lyapunov index and the sum of index is less than zero. This proves it is a hyperchaotic system.

2) Making palette

The process of making palette is divided into following steps:

Steps1: A white image is generated and the compression ratio of secrete image depends on the size of image. Let A represents the generated white image: $A = \{a(i,j,k), 1 \le i \le Ma, 1 \le j \le Na, k = 1,2,3\}$, in which $a(i,j,1)$, $a(i,j,2)$ and $a(i,j,3)$ represent three basic color components——red, green, blue of some pixel. The value of all components is 255 because the image is white image. The coding bits of each encoded pixel of secret image are $\lceil \log_2(Ma \times Na) \rceil$, in which $\lceil \cdot \rceil$ represents upward rounding.

Steps2: Original value is set as x_0^1, y_0^1, z_0^1 and u_0^1. Hyperchaotic sequence *x*, *y*, *z* and *u* are generated by equation (1) and the statistic distribution laws and value range of *x*, *y*, *z* and *u* change by equation (2). The sequences are:

$$X = \{x(i), 1 \le i \le Lx, x \in [0,255]\},$$
$$Y = \{y(i), 1 \le i \le Ly, y \in [0,255]\},$$
$$Z = \{z(i), 1 \le i \le Lz, z \in [0,255]\}.$$
$$X(i) = \mathrm{mod}(\lfloor |x(i) + x(i + N/2)| \times M \rfloor, 256) \quad (2)$$

Steps3: Palette *A*' is calculated by equation (3) utilizing sequences X, Y, Z with length of $Ma \times Mb$ and each component of white image:

$$A' = \{a'(i,j,k), 1 \le i \le Ma, 1 \le j \le Na, k = 1,2,3\}$$
$$\begin{cases} a'(i,j,1) = a(i,j,1) \oplus x((i-1)Na + j) \\ b'(i,j,1) = b(i,j,1) \oplus y((i-1)Na + j) \\ c'(i,j,1) = c(i,j,1) \oplus z((i-1)Na + j) \end{cases} \quad (3)$$

B. Search for index value

S represents the secret image:

$S = \{s(i,j,k), 1 \le i \le Ms, 1 \le j \le Ns, k = 1,2,3\}$, in which $s(i,j,1)$, $s(i,j,2)$, $s(i,j,3)$ represent three basic color components——red, green, blue of some pixel. The Euclidean distance between the pixel of secret image (i,j) and the color of palette is calculated by Equation (4).

$$D(m,n) = \sqrt{\sum_{k=1}^{3} (s(i,j,k) - a'(m,n,k))^2} \quad (4)$$

The code value B of pixel point s(i,j) of secret image is inquired by encoding the value m and n which is obtained when Euclidean distance is minimum: $B = \{b(i), 1 \le i \le Ms \times Ns \times \lceil \log_2(Ma \times Na) \rceil, b(i) \in \{0,1\}\}$.

(3) Code encryption

Code B is divided into H sections represented as $B_1, B_2 \ldots, B_h \ldots, B_H$, in which $B_h = \{b_h(i), 1 \le i \le L, b_h(i) \in \{0,1\}\}$ and the length of each section is L. Initial key SK is constructed from sequence v of length （H+1）×L, and it is formed by hyperchaotic sequence x, y, z, u generated by initial value x_0^2, y_0^2, z_0^2 and u_0^2: $SK = \{sk(i), 1 \le i \le (H+1) \times L, sk(i) \in \{0,1\}\}$. Construcetion method is showed by equation (5):

$$\begin{cases} sk(i) = 0 & if\ v(i) < 0 \\ sk(i) = 1 & if\ v(i) \ge 0 \end{cases} \quad (5)$$

Original key divided into H+1 section represented as $SK_0, SK_1 \ldots, SK_h, \ldots, SK_H$, in which $SK_h = \{sk_h(i), 1 \le i \le L, sk_h(i) \in \{0,1\}\}$ and the length of each section is L.

The code generated by first-round encryption is represented as $B_1^1, B_2^1, \cdots, B_h^1, \cdots, B_H^1$, in which $B_h^1 = \{b_h^1(i), 1 \le i \le L, b_h^1(i) \in \{0,1\}\}$. The encryption key of first round is represented as $SK_1^1, SK_2^1, \cdots, SK_h^1, \cdots, SK_H^1$, in which $SK_h^1 = \{sk_h^1(i), 1 \le i \le L, sk_h^1(i) \in \{0,1\}\}$. The encryption algorithm is showed by equation (6):

$$B_h^1 = B_h \oplus SK_h^1 \quad (6)$$

Key is calculated by equation (7):

$$\begin{cases} sk_h^1(i) = sk_0(i) & if\ h = 1 \\ sk_h^1(i) = sk_{h-1}(i) \oplus b_{h-1}^1(L) & if\ i = 1\ and\ h \ne 1 \\ sk_h^1(i) = sk_{h-1}(i) \oplus b_{h-1}^1(i-1) & if\ i \ne 1\ and\ h \ne 1 \end{cases} \quad (7)$$

The code in first section is encrypted by initial key SK_0. The keys of code in other sections are generated by iterating initial key, vertical cyclic shift (physical space cycle) and horizontal cyclic iteration (time space cycle). The generated code of K^{th} round encryption is represented as $B_1^k, B_2^k \cdots, B_h^k \cdots, B_H^k$, in which $B_h^k = \{b_h^k(i), 1 \le i \le L, b_h^k(i) \in \{0,1\}\}$. The key of k^{th} round encryption is represented

as $SK_1^k, SK_2^k \cdots, SK_h^k \cdots, SK_H^k$, in which $SK_h^k = \{sk_h^k(i), 1 \le i \le L, sk_h^k(i) \in \{0,1\}\}$. Encryption algorithm is showed in equation (8):

$$B_h^k = B_h^{k-1} \oplus SK_h^k \tag{8}$$

Key of k[th] round encryption is calculated by Equation (9).

$$\begin{cases} sk_h^k(i) = sk_H(i) \oplus b_H^{k-1}(L) & if\ i = 1\ and\ h = 1 \\ sk_h^k(i) = sk_H(i) \oplus b_H^{k-1}(i-1) & if\ i \neq 1\ and\ h = 1 \\ sk_h^k(i) = sk_{h-1}(i) \oplus b_{h-1}^k(L) & if\ i = 1\ and\ h \neq 1 \\ sk_h^k(i) = sk_{h-1}(i) \oplus b_{h-1}^k(i-1) & if\ i \neq 1\ and\ h \neq 1 \end{cases} \tag{9}$$

Key of code in first section is generated by XOR between the original key and the shifted encryption data in last section during previous encryption process. Keys of code in other sections are generated by XOR between original password and cyclic shifted encryption code in previous section in physical space.

III. SIMULATION ANALYSIS

The simulation experiments are conducted in Matlab7.0. Several secret images are chosen to simulate in this experiment. The result showed that the algorithm is feasible. The following experiment utilizes 130×130 24 bits true color image "sudayingyong.jpg" as secret image. The size of white image is 64×64. In equation (4), the system parameters are a=35, b=3, c=35, d=8, and the initial value are 2.5, 5.2, 3.0, 7.3. Runge-Kutta4-5 method is utilized and the stepping is 0.001. In equation (2), M is chosen as 3×10^4. Original white image and palette are shown as Figure2 (a) (b).

Original secret image and reconstructed image after compression are shown as Figgure 3 (a) (b).

A. Analysis of encryption effect

The encryption process can be divided into two processes in the study. One is compress coding process. It is the process of searching subscript corresponding to each point of secret image in palette and coding. The decryption image can be obtained by reconstructing the code. Simulation experiment will analyze the compression effect and the quality of reconstructed image in the process. Another process is to circularly encrypt code. It will continue to be encrypted by loop feedback in physical space and time space. The encrypted secret image can be obtained by directly reconstructing encrypted code. The encryption result is analyzed, the pixel distribution histogram after encryption is drawn, the correlation coefficients of encrypted image in all direction are calculated, and the similarity degree between plaintext and ciphertext, information entropy, peak signal to noise ratio etc. are calculated.

1) Analysis of compression ratio and quality of reconstructed image

The compression effect can be represented by compression ratio. And compression ratio can be calculated by equation (10).

$$CR = 1 - \frac{\text{image data after compression}}{\text{image data before compression}} \times 100\% \tag{10}$$

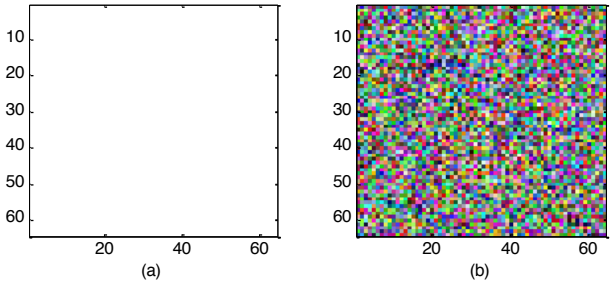

Figure 2. Original white image and palette

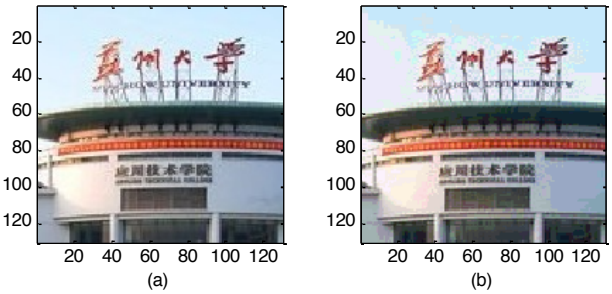

Figure 3. Original secret image and reconstructed secret image after compression

Compression ratio is zero when bits are equal before and after the image is compressed.

The quality of reconstructed image can be described as the similarity degree between reconstructed image and original image. The equation (11) can be utilized to calculate similarity degree of color image.

$$SIM = \frac{\sum_{i=1}^{Ms}\sum_{j=1}^{Ns}\left(\sum_{k=1}^{3}(s(i,j,k)-\overline{s(k)})(s'(i,j,k)-\overline{s'(k)})\right)}{\sqrt{\sum_{i=1}^{Ms}\sum_{j=1}^{Ns}\left(\sum_{k=1}^{3}(s(i,j,k)-\overline{s(k)})^2\right)}\sqrt{\sum_{i=1}^{Ms}\sum_{j=1}^{Ns}\left(\sum_{k=1}^{3}(s'(i,j,k)-\overline{s'(k)})^2\right)}} \tag{11}$$

where $\overline{s(k)} = \frac{1}{Ms \times Ns}\sum_{i=1}^{Ms}\sum_{j=1}^{Ns} s(i,j,k)$,

$\overline{s'(k)} = \frac{1}{Ms \times Ns}\sum_{i=1}^{Ms}\sum_{j=1}^{Ns} s'(i,j,k)$, k=1,2,3

Where s and s' represent original secret image and secret image after reconstruction. The similarity degree is 1 when s and s' are equal.

The similarity degree between reconstructed image and original image can also be calculated with the change of palette size and the compression ratio of encryption image. The relation between compression ratio and similarity degree is shown as Figure.4.

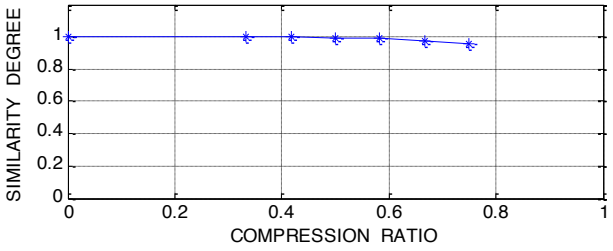

Figure 4. The relation between compression ratio and similarity degree

Figure 4 shows that with the increase of compression ratio, the similarity degree between reconstructed image and original image becomes lower, although small. The similarity degree between reconstructed image and original image is 0.99962 when the compression ratio is 0.33333. The similarity degree is 0.95316 when the compression ratio is 0.75000.

2) Effect of cyclic encryption

a) Correlation coefficient

The decrypted secret image s' and encrypted secret image s'' can be obtained by utilizing codes after compression coding and codes after loop feedback to index from palette and reconstruct secret image.

The equation (12) can be utilized to calculate original image s, and correlation coefficients of adjacent pixel of three components—red, green, blue in all directions s', s''. The result is shown as Table I.

TABLE I. COMPARISON OF ENCRYPTED IMAGE

Image		Horizontal Direction	Vertical Direction	Diagonal Direction
Original Image	Red Component	0.9471	0.9088	0.8711
	Green Component	0.9507	0.9151	0.8786
	Blue Component	0.9572	0.9242	0.8924
	Average Value	0.9517	0.9160	0.8807
Reconstructed Image	Red Component	0.9451	0.9059	0.8691
	Green Component	0.9491	0.9131	0.8768
	Blue Component	0.9536	0.9185	0.8856
	Average Value	0.9493	0.9125	0.8772
Encrypted Image	Red Component	0.0040	-0.0082	-0.0086
	Green Component	-0.0074	0.0129	0.0032
	Blue Component	-0.0014	0.0012	0.0142
	Average Value	-0.0016	0.0020	0.0029

$$r_{xy} = \frac{|\mathrm{cov}(x,y)|}{\sqrt{D(x)}\sqrt{D(y)}} \qquad (12)$$

In equation (12):

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2 \quad , \quad E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \quad ,$$

$$\mathrm{cov}(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - y(x)) \quad .$$

Table I shows that the change of correlation between original image and reconstructed image after compression and coding is small in three directions. The quality of compressed image changes little. The correlation coefficient of adjacent pixel in image approaches to zero after loop encryption. Statistical characteristics of original image have been diffused to ciphertext randomly.

b) Visual effect and gray distribution

Reconstructed image s' and encryption image s'' can be shown as Figure 5 (a) (b). The histogram of original image s and red component of encryption image s'' are shown as Fgure 6 (a) (b).

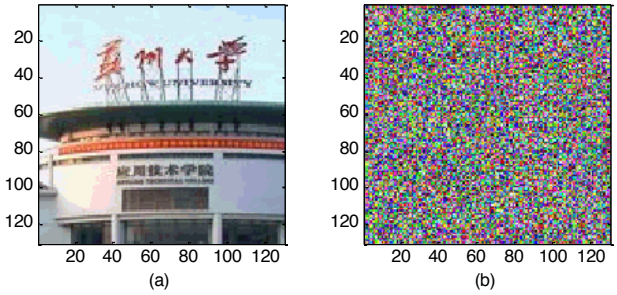

Figure 5. Reconstructed image and encryption image

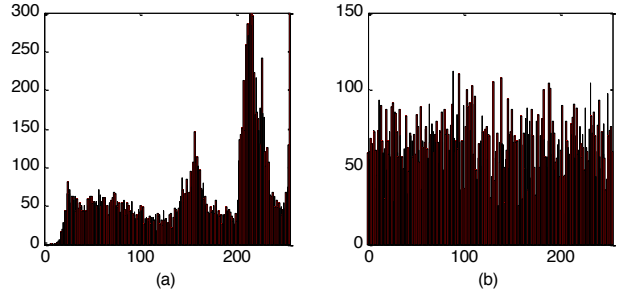

Figure 6. Original image and red component histogram of encrypted image

Figure 5 and Figure 6 shows that encrypted image totally change visual effect, and the gray distribution of image tends to be uniform.

c) Peak signal to noise ratio

The calculation method on peak signal to noise ratio in literature [15] can be a reference, and equation (13) can be utilized to calculate peak signal to noise ratio.

Peak signal to noise ratio:

$$PSNR = 10\log_{10}\left(\frac{XYZ\max_{x,y,z} S^2_{x,y,z}}{\sum_{x,y,z}(S_{x,y,z} - S'_{x,y,z})^2}\right) \qquad (13)$$

where $S_{x,y,z}$ stands for pixel point in original secret image, $S'_{x,y,z}$ stands for pixel point in encrypted image. x, y are row and column respectively, while the value of z are 1,2,3. The encrypted image is undecipherable when peak signal to noise ratio is blow 20 dB. The peak signal to noise ratio of encrypted image S'' by simulation calculation is 7.1862. It proves that the effect of encryption is good and the plaintext can be concealed totally. The peak signal to noise ratio of reconstructed image S' is 33.3927, far more than 20. It proves that the quality of compressed image is good.

d) Information entropy

Equation (14) is utilized to calculate information entropy of color image. It calculates information entropies of three primary color components and obtains the average value as the information entropy.

$$H = -\frac{1}{3}\sum_{i=1}^{3}\sum_{j=0}^{2^8-1} p(i,j)\log_2 p(i,j) \qquad (14)$$

Where i represents the i^{th} primary color component in three primary color components, j represents the j^{th} gray value in 256 gray levels in experiments. The greater the

information entropy, the more uniform the gray level distributes in image. Its maximum value is 8. The simulation result shows that the information entropy of original image is 7.2611, and the information entropy of encrypted image is 7.9416. It is close to 8 and it is uniformly distributed.

B. *Key space and sensitivity analysis*

1) Key space

The hyperchaotic system showed by equation (4) has 4 state variables. The hyperchaotic sequences for making palette are generated by original value x_0^1, y_0^1, z_0^1 and u_0^1. The hyperchaotic sequence for code loop decryption are generated by different original value x_0^2, y_0^2, z_0^2 and u_0^2. If original value is expressed by 15 bits double-precision real number, the key space can reach $10^{4\times15}$ or $10^{4\times15}$ corresponding to the length of 199 or 399 bits.

2) Key sensitivity

In order to test key sensitivity, key x_0^1 is changed 10^{-15}. Then correct decryption and error decryption are simulated. The result is showed in (a) and (b) of Figure7.

Let s and s' express the image of correct decryption and error decryption. Equation (15) is utilized to calculate mean square error of color image based on the calculation method of mean square error provided by literature [16].

$$EMS = \frac{1}{Ms \times Ns \times 3} \sum_{i=1}^{Ms} \sum_{j=1}^{Ns} \sum_{k=1}^{3} \{(s(i,j,k) - s'(i,j,k)\}^2 \quad (15)$$

The mean square error of image of correct decryption and error decryption is 12236.5585.

IV. CONCLUSION

The encryption and compression scheme for color image is proposed to realize encryption and compression simultaneously. Hyperchaotic system is utilized to make palette with fruit colors and uniformly distributed color. It can effectively reduce the quality loss of reconstructed image after compression. By changing the size of white image, the compression ratio of secret image can be adjusted. The similarity degree of reconstructed image and original image is above 95% when the compression ratio reaches 75%. The initial key is generated by hyperchaotic system to further circularly encrypt codes that have been encrypted and compressed. Encrypted image have a good visual effect, large key space. Besides, adjacent pixel correlation coefficient, peak signal to noise ratio, information entropy and other evaluation index have reached or better than the encryption algorithm proposed by literature [15]. The essence of obtaining a good compression effect is the design of palette. To improve the design method of palette will help to improve the compression quality of color image.

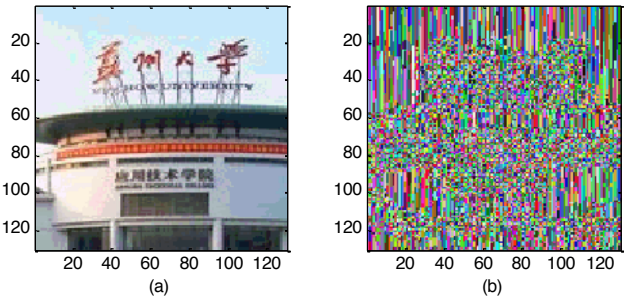

Figure 7. Image of correct decryption and error decryption

REFERENCE

[1] Tanha Mehrdad, Kheradmand Reza, Ahmadi-Kandjani Sohrab. Gray-scale and color optical encryption based on computational ghost imaging,Applied Physics Letters,Vol. 101, Issue 10,2012, pp. 101108-101108-3.

[2] Seyedzadeh S.M., Moosavi S.M.S., Mirzakuchaki S. Using self-adaptive coupled piecewise nonlinear chaotic map for color image encryption scheme, 19th ICEE, 2011, pp.1- 6.

[3] Sathishkumar G.A., Ramachandran S., Bhoopathy Bagan K. Image encryption using random pixel permutation by chaotic mapping, ISCI, 2012, pp.247- 251.

[4] Sahami S., Shayesteh M.G. Bi-level image compression technique using neural networks. Image Processing, Vol. 6, Issue 5, 2012, pp.496 -506.

[5] Chopra G., Pal A.K. An Improved Image Compression Algorithm Using Binary Space Partition Scheme and Geometric Wavelets. IEEE Transactions on Image Processing, Vol. 20, Issue 1, 2011, pp.270-275. http://dx.doi.org/10.1109/TIP.2010.2056378

[6] Hong-Sik Kim, Joohong Lee, Hyunjin Kim, Sungho Kang, Woo Chan Park . A Lossless Color Image Compression Architecture Using a Parallel Golomb-Rice Hardware CODEC, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 21, Issue 11, 2011, pp. 1581- 1587.

[7] Li Juan, Feng Yong, Yang Xuqiang. 3D Chaotic Encryption Scheme for Compressed Image. ACTA OPTICA SINICA, Vol. 30, Issue 2, 2010, pp. 399-404. http://dx.doi.org/10.3788/AOS20103002.0399

[8] Monica D.N., Shunmuganathan K.L., Suresh L.P. Efficient compression of encrypted grayscale images . ICSCCN. 2011,pp.564 – 568.

[9] Xinpeng Zhang. Lossy Compression and Iterative Reconstruction for Encrypted Image. IEEE Transactions on Information Forensics and Security, Vol. 6, Issue 1, 2011, pp. 53- 58. http://dx.doi.org/10.1109/TIFS.2010.2099114

[10] WEN Chang-ci,WANG Qin, CHEN Qing, YUAN Zhi-shu, MIAO Xiao-ning. Novel encryption for JPEG color image.Systems Engineering and Electronics, Vol. 34, Issue 6, 2012, pp. 1283-1287.

[11] Yang Hua-Qian,Liao Xiao-Feng,Kwok-Wo Wong, Zhang Wei, Wei Pengcheng.SPIHT-based joint image compression and encryption. Acta Phys. Sin. , Vol. 61, Issue 4, 2012, pp. 040505-1-040505-8.

[12] Ching-Hung Yuen, Oi-Yan Lui, Kwok-Wo Wong. Application of chaotic maps for simultaneous lossy image compression and encryption . Circuits and Systems (ISCAS). 2012. pp.393-396.

[13] El-Latif A.A.A., Li Li, Ning Wang, Xiamu Niu. Image Encryption Scheme of Pixel Bit Based on Combination of Chaotic Systems. (IIH-MSP). 2011. pp.369 -373.

[14] Bhatnagar G., Wu Q.M.J. Chaos-Based Security Solution for Fingerprint Data During Communication and Transmission. IEEE Transactions on Instrumentation and Measurement, Vol. 61, Issue 4, 2012, pp. 876 -887. http://dx.doi.org/10.1109/TIM.2011.2179330

[15] WEN Chang-ci, WANG Qin, HUANG Fu-min, YUAN Zhi-shu, TAO Chun-sheng. Self-adaptive encryption algorithm for image based on affine and composed chaos. Journal on Communications, Vol. 33, Issue 11, 2012, pp. 119-127.

[16] Wang Jing, Jiang Guo- Ping .Cryptanalysis of a hyper- chaotic image encryption algorithm and its improved version. Acta Phys. Sin. , Vol. 60, Issue 6, 2011, pp. 060503- 1-060503- 11.

[17] Zhu Cong-xu, Hu Yu-ping ,Sun Ke-hui. New Image Encryption Algorithm Based on Hyper chaotic System and Ciphertext Diffusion in Crisscross Pattern. Journal of Electronics & Information Technology , Vol. 34, Issue7, 2012, pp. 1735-1743.

[18] Liu Ming-hua, Feng Jiu-chao. A new hyper chaotic system. ACTA PHYSICA SINICA , Vol. 58, Issue7, 2009, pp. 457-4462.

AUTHORS

PENG Jing-yu is with the College of Applied Technique, Soochow University, Suzhou, China.

GONG Sheng-rong is with College of Computer Science and Technology, Soochow University, Suzhou, China.

This article is an extended and modified version of a paper presented at the 2012 International Conference on Artificial Intelligence and Its Application in Industry Production (AIAIP 2012), held in Wuhan, China in December 2012. Manuscript received 31 May 2013. Published as resubmitted by the authors 26 June 2013.