A Digital Image Encryption Algorithm Based A Composition of Two Chaotic Logistic Maps

Ismail Amr Ismail[1], Mohammed Amin[2], and Hossam Diab[2]
(Corresponding author: I. A. Ismail)
Faculty of Computers and Informatics, Zagazig University, Egypt[1]
Faculty of Science, Menoufia University, Egypt[2] (Email: hossamdiab_86@yahoo.com)
(Received Oct. 29, 2006; revised and accepted June 15, 2007 & May 14, 2008)

Abstract

The chaos based cryptographic algorithms have suggested several advantages over the traditional encryption algorithms such as high security, speed, reasonable computational overheads and computational power. This paper introduces an efficient chaos-based stream cipher, composing two chaotic logistic maps and a large enough external secret key for image encryption. The external secret key is used to derive the initial conditions for the chaotic maps, and is employed with the two chaotic maps to confuse the relationship between the cipher image and the plain image. In the encryption phase, the pixels are encrypted using an iterative cipher module based feedback and data-dependent inputs mechanism for mixing the current encryption parameters with previously encrypted information. To make the cipher more robust against any attack, the secret key is modified after encryption of each pixel of the plain image. The results of several experimental, statistical analysis and key sensitivity tests show that the proposed image encryption scheme provides an efficient and secure way for real-time image encryption and transmission.

Keywords: Chaos cryptography, data security, decryption, image encryption, logistic maps, statistical analysis

1 Introduction

The advent of personal computers and the Internet has made it possible for anyone to distribute worldwide digital information easily and economically. Many applications like military image databases, confidential video conferencing, medical imaging system, cable TV, online personal photograph album, etc. require reliable, fast and robust security system to store and transmit digital images. In this environment, there are several security problems associated with the processing and transmission of digital images over an open network: It is necessary to assure the confidentiality, the integrity and the authenticity of the digital image transmitted. Also, Encryption of images is different from that of text due to some intrinsic features of images such as redundancy of data, strong correlation among adjacent pixels, being less sensitive as compared to the text data i.e. a tiny change in the attribute of any pixel of the image does not drastically degrade the quality of the image and bulk capacity of data. To meet these challenges, a wide variety of cryptographic protocols have been appeared in the scientific literature [3, 4, 5, 6, 7, 9, 11, 12, 14, 15, 17, 18, 20, 21, 22]. Traditional data cryptosystems like DES, 3DES, AES and RSA exhibit some drawbacks and weakness [1, 8, 13] in the encryption of digital images (for example, a large computational time and high computing power are needed, so, these techniques suffer from low-level efficiency when the image is large.), consequently, they are not suitable for image encryption. In this respect, chaos based encryption techniques are considered good for practical use as chaos based algorithms provide a good combination of speed, high security, complexity, reasonable computational overheads and computational power. Moreover, chaos-based and other dynamical systems based algorithms have many important properties such as the sensitive dependence on initial conditions and system parameters, pseudo random properties, ergodicity, non-periodicity. These properties meet some requirements such as sensitive to keys, diffusion and mixing in the sense of cryptography. Therefore, chaotic dynamics are expected to provide a fast and easy way for building superior performance cryptosystems.

A number of chaos based image encryption scheme have been developed in recent years. We discuss some of them in brief. Fridrich [12] demonstrated the construction of a symmetric block encryption technique based on two-dimensional standard baker map. There are three basic steps in Fridrich's method: (a) Choose a chaotic map and generalize it by introducing some parameter, (b) Discriminate the chaotic map to a finite square lattice of points that represent pixels, (c) Extend the discriminated map to three-dimensions and further compose it with a simple diffusion mechanism. Scharinger [18] designed a chaotic based image encryption technique, in which whole image is taken as a single block and which is permuted through a key-controlled chaotic system. In addition, a

shift register pseudo random generator is also adopted to introduce the confusion in the data. Yen and Guo [20] proposed an encryption method called BRIE based on chaotic logistic map. The basic principle of BRIE is bit recirculation of pixels, which is controlled by a chaotic pseudo random binary sequence. The secret key of BRIE consists of two integers and an initial condition of the logistic map. Yen and Guo [22] also proposed an encryption method called CKBA (Chaotic Key Based Algorithm) in which a binary sequence as a key is generated using a chaotic system. The image pixels are rearranged according to the generated binary sequence and then XORed and XNORed with the selected key. Li and Zheng [14] pointed out some defects in the encryption schemes presented in references [20, 22] and discussed some possible improvements on them. Chen et al. [6] have proposed a symmetric image encryption in which a two-dimensional chaotic map is generalized to three dimension for designing a real time secure image encryption scheme. This approach employs the three-dimensional cat map to shuffle the positions of the image pixels and uses another chaotic map to confuse the relationship between the encrypted and its original image. Zhao and Chen [23] introduced the concept of ergodic matrix, and used it to uniformly present scramble algorithms based on pixel shifting. However, there are still some potential weak points existing in this pure position permutation algorithm, which is frail under known-text attack. Guan et al. [10] presented an image encryption scheme, in which shuffling the positions and changing the grey values of image pixels are combined to confuse the relationship between the cipher-image and the plain-image. Firstly, the Arnold cat map is used to shuffle the positions of the image pixels in the spatial-domain. Then the discrete output signal of the Chen's chaotic system is preprocessed to be suitable for the grayscale image encryption, and the shuffled image is encrypted by the preprocessed signal pixel by pixel. Wang et al. [19] proposed a successful chosen-plain-text cryptanalytic attack (With the knowledge of some specially designed plain-images, the equivalent initial condition of diffusion process is calculated and a valid equivalent 3D Cat matrix is built.) which is composed of two mutually independent procedures: the cryptanalysis of the diffusion process and the cryptanalysis of the spatial permutation process. Pareek et al. [16] proposed an image encryption scheme in which an external secret key and two chaotic logistic maps are employed. The initial conditions for the both logistic maps are derived using the external secret key and eight different types of operations are used to encrypt the pixels of an image.

In the next section, we propose an efficient chaos-based feedback stream cipher for image encryption satisfying the requirements of secure image transfer. The paper is organized as follows. Section 2 is devoted to the proposed chaos based image encryption algorithm. The experimental results and the security analysis of the proposed scheme are presented in Sections 3 and 4. Finally, Section 5 gives the conclusion.

2 The Proposed Image Encryption Algorithm

The following proposed cryptosystem is a symmetric key steam cipher algorithm, which utilizes the essence of chaos, i.e., sensitivity on the initial condition as well as on system parameter. This cryptosystem does not use explicitly the system parameter or initial condition of the chaotic map (logistic map) as a secret key. However, these parameters are generated by an external secret key. The cryptosystem is further made robust against any reasonable attack by using the feedback technique, i.e., encryption of each pixel of the plain-image is also made dependent upon the encryption properties of the previous pixel of the plain-image. Further, new features of the proposed stream cipher include the heavy use of data-dependent inputs, data-dependent iterations (variable number of iterations for each map depending on the current value of the key, the value of the previous cipher pixel and the output of the logistic map).

The proposed procedure consists of three steps: choosing a chaotic map, digitization and key schedule.

2.1 Choosing A Chaotic Map

Choosing maps for encryption algorithms is not an easy task and one should consider only maps with the following properties: mixing property, robust chaos and large parameter set [2].

- Mixing property: Mixing property of chaotic maps is closely related to property of diffusion in encryption transformations (algorithms). If we think of the set of possible (sensible) plaintexts as an initial region in the phase space of the map (transformation), then it is the mixing property (or in other terms, sensitivity to initial conditions) that implies "spreading out of the influence of a single plaintext digit over many ciphertext digits".

- Robust chaos: A good encryption algorithm spreads also the influence of a single key digit over many digits of ciphertext. The keys of an encryption algorithm represent its parameters. Therefore, we should consider only such transformations in which both parameters and variables are involved in a sensitive way.

- Parameter set: One should consider only systems that have robust chaos for large set of parameters (keys). larger parameter space of the dynamical system implies that its discriminated version will have larger K. In this paper, we design the proposed cipher using logistic chaotic maps.

$$x(n+1) = 4x(n)[1 - x(n)]. \quad (1)$$

2.2 Digitization

Digitization is a process in which the map G: $y \rightarrow y$ is replaced with the map $F : x \rightarrow x$. Digitization is not

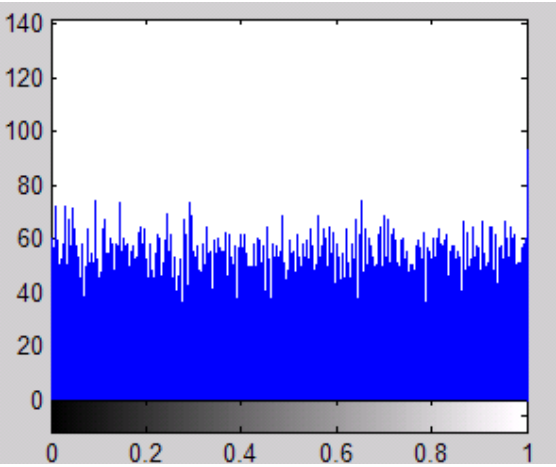

Figure 1: Histogram of digitized sequence

a unique process. However, in many cases one can identify "a natural way" in doing this. Thus, for example, if $\beta = \{c_0,\ldots,c_{2m-1}\}$ is a finite partition of the phase space y, then $x = \{0,\ldots,2^m - 1\}$ and F is the restriction of G on x (assuming that such restriction exists).

In our work, we make digitization of the logistic map as follows:

- **First method.** Firstly, the chaotic sequence $\{X_k\}$ is generated through Equations (1), which has to be amplified by a scaling factor (10^4) and round off to integer-sequence according to Equations (2).

$$Z_k = round((x_k * 10^4) \bmod 256). \quad (2)$$

- **Second method.** The chaotic sequence $\{X_k\}$ is generated through Equations (1), which has to be amplified by a scaling factor (2^8-1) and round off to integer-sequence $\{z_k\}$ according to equation

$$z_k = round((x_k * (2^8 - 1) \bmod 256). \quad (3)$$

This transformation implies that, when the randomly generated chaotic sequence (input values) is uniformly distributed, the output of the digitization process is also uniformly distributed Figure 1.

2.3 Key Schedule

In view of the basic need of cryptology, the cipher-text should have close correlation with the key. There are two ways to accomplish this requirement: one is to mix the key thoroughly into the plain-text through the encryption process, another is to use a good key generation mechanism. We use a 104-bit external secret key to derive a vector of 5 parameters as initial conditions of the system in addition to its use in each pixel encryption. Now, we derive the initial condition from the external key:

- The image encryption process utilizes an external secret key of 104-bit long. Further, the secret key is divided into blocks of 8-bit

$$K = k_1 k_2 \ldots k_{26}, (inhexadecimal). \quad (4)$$

Here, $k_i's$ are the alphanumeric characters (0-9 and A-F). Alternatively, the secret key can be represented in ASCII mode as

$$K = k_1 k_2 \ldots k_{13} (inASCII). \quad (5)$$

- To calculate the initial conditions (L is a floating number in (0,1) , and S is an integer to be used as a seed) for the first logistic map, divide the key into three groups of four blocks as follows:
  - First group: k_1, k_2, k_3, k_4,
  - Second group: k_5, k_6, k_7, k_8,
  - Third group: k_9, k_{10}, k_{11}, k_{12}.

For each group we calculate:

$$\begin{aligned} g_i &= \sum_{j=1}^{j=4} k_j * 10^{-J}, i = 1,2,3, \\ R &= \prod_{i=1}^{3} g_i \bmod 1. \end{aligned}$$

Where the notation ($x \bmod 1$) stands for the fractional parts of a real number (x) by subtracting or adding an appropriate integer number. Now, the value of L is

$$L = (R + \frac{k_{13}}{256}) \bmod 1. \quad (6)$$

And the value of S is

$$S = round(\sum_{i=1}^{3} g_i * 10^4 + L * 10^4) \bmod 256. \quad (7)$$

- To calculate the initial conditions (L', and S') for the second logistic map,

$$\begin{aligned} V_1 &= \sum_{i=1}^{13} K_i, V_2 = \bigoplus_{i=1}^{13} k_i, \\ V &= \frac{V_2}{V_1}. \end{aligned}$$

Now, the value L' is

$$L' = (V + \frac{k_{13}}{256}) \bmod 1,$$

and the value of S' is

$$S' = round(V_1 + V_2 + L' * 10^4) \bmod 256, \quad (8)$$

and the initial value for the diffusion process C_0 is

$$C_0 = round((L * l' * 10^4)) \bmod 256. \quad (9)$$

2.4 The Scheme

Encryption and decryption procedures can be depicted as follows:

Step 1. Use external secret key of 104-bit long and divide it into blocks of 8-bit as in Equations (4), (5).

Step 2. Use two chaotic logistic maps,

$$x(n+1) = 4x(n)[1-x(n)], \quad (10)$$
$$y(n+1) = 4y(n)[1-y(n)]. \quad (11)$$

And compute the initial conditions L, S (for the map X) and L', S'(for the map y) as in Equations (6), (7), (8), and (9), respectively.

Step 3. If the obtained values form Equations (10), (11) are within the subinterval (0.2, 0.8), then go to Step 4; otherwise, keep Step 2 in execution, iterating the map, until a desired number in (0.2, 0.8) is obtained.

Step 4. Digitize the values x and y obtained from the logistic maps by amplifying them with a proper scaling and sampling as in Equations (2), (3). The digitized values are designated as $\phi(n)$ and $\phi'(n)$, respectively.

Step 5. Calculate the intermediate value C_1 and C_2 as follows:

$$\begin{aligned} C_1(n) &= \phi(n) \oplus \{[k_1(n) + \phi(n)] \bmod N\} \\ &\quad \oplus\{[C_1(n-1) + k_2(n) \bmod N]\}, \\ C_2(n) &= \phi'(n) \oplus [k_3(n) + \phi'(n)] \bmod N \\ &\quad \oplus[C_2(n-1) + k_4(n) \bmod N]. \end{aligned}$$

Where $C_1(n-1)$ and $C_2(n-1)$ are the previously output; $C_1(0) = S$, $C_2(0) = S'$; and N is the color level (for a 256 grey scale image, $N = 256$).

Step 6. Compute the ciphered pixel as,

$$\begin{aligned} C(n) &= \{[k_5(n) + C_1(n))] \bmod N\} \\ &\quad \oplus\{[k_6(n) + C_2(n))] \bmod N\} \\ &\quad \oplus\{[k_7(n) + I(n))] \bmod N\} \\ &\quad \oplus\{[k_8(n) + C(n-1))] \bmod N\}. \end{aligned}$$

Where $I(n)$ is the currently operated pixel, $C(n-1)$ is the previously output cipher-pixel, and $C(0)$ be an initial value that is computed according to Equations (9).

Step 7. Update, after encryption of each pixel, the key and the coming logistic maps inputs as follows:

$$\begin{aligned} k_{13} &= k_{13}^0 \oplus C(n) \\ k_i &= (k_i^0 + k_{13}) \bmod 256, 1 \le i \le 12 \\ x &= (x_0 + \frac{C(n)}{256} + \frac{k_9}{256} + \frac{k_{10}}{256}) \bmod 1, \\ y &= (y_0 + \frac{C(n)}{256} + \frac{k_{11}}{256} + \frac{k_{12}}{256}) \bmod 1. \end{aligned}$$

where x_0 and y_0 are the computed initial values (L and L'), and k_i^0 is the initial external secret key.

To this end, the process of decryption is completely similar to the encryption process described above, only the difference would be in the Step6, (using the inverse transform of Step6). Decryption of each cipher-image pixel to produce its corresponding plain-image pixel can be expressed mathematically as:

$$\begin{aligned} I(n) &= \{\{[k_5(n) + C_1(n))] \bmod N\} \\ &\quad \oplus\{[k_6(n) + C_2(n))] \bmod N\} \\ &\quad \oplus\{[k_8(n) + C(n-1))] \bmod N\} \\ &\quad \oplus\{C(n)\} + \{N - k_7(n))\}\} \bmod N. \end{aligned}$$

We note that, the above cipher can be adopted to encrypt grayscale and color images. For color images, first decompose the color image into *R-G-B* components. Second, encrypt each component (*R-G-B*) separately by the proposed algorithm. Finally, concatenate the encrypted components together to get the encrypted color image.

3 Experimental Results

Some experimental results are given in this section to demonstrate the efficiency of the proposed scheme. In our experimental results, several images are evaluated. For the evaluations of encryption quality, the correlation coefficient (C.C) is used which can be calculated by [6, 16],

$$C.C = \frac{N\sum_{j=1}^{N}(x_j \times y_j) - \sum_{j=1}^{N} x_j \times \sum_{j=1}^{N} y_j}{\sqrt{(N\sum x_j^2 - (\sum_{j=1}^{N} x_j)^2) \times (N\sum y_j^2 - (\sum_{j=1}^{N} y_j)^2)}} \quad (12)$$

where x and y are grey scale pixel values of the original and encrypted images. A striking example of the degree to which the proposed cipher can reveal patterns in the plaintext is shown in Figure 2, where the plain images are encrypted by the secret key "32C69C29084AF8F23AF4AD0E99" (hexadecimal).

The results are also compared with the encryption scheme presented by **Guan** et al. in [10], as abbreviation Guan, and an encrypted image generated by the Chen et al. in [6], as abbreviation **Chen**. Obviously, the proposed method hides all features of the original image where the ciphered image is significantly different from its plain image, i.e., there is no visual information observed in the encrypted images, and the encrypted images are visual indistinguishable. Computationally, it clear that there is negligible correlation between the plain image and ciphered image, Table 1, where the proposed scheme retains the smallest correlation coefficients (C.C). Thus, the proposed scheme outperforms both Guan method and Chen method.

4 Security Analysis

A good encryption scheme should resist all kinds of known attacks, such as known-plain-text attack, ciphertext only attack, statistical attack, and various brute force attacks

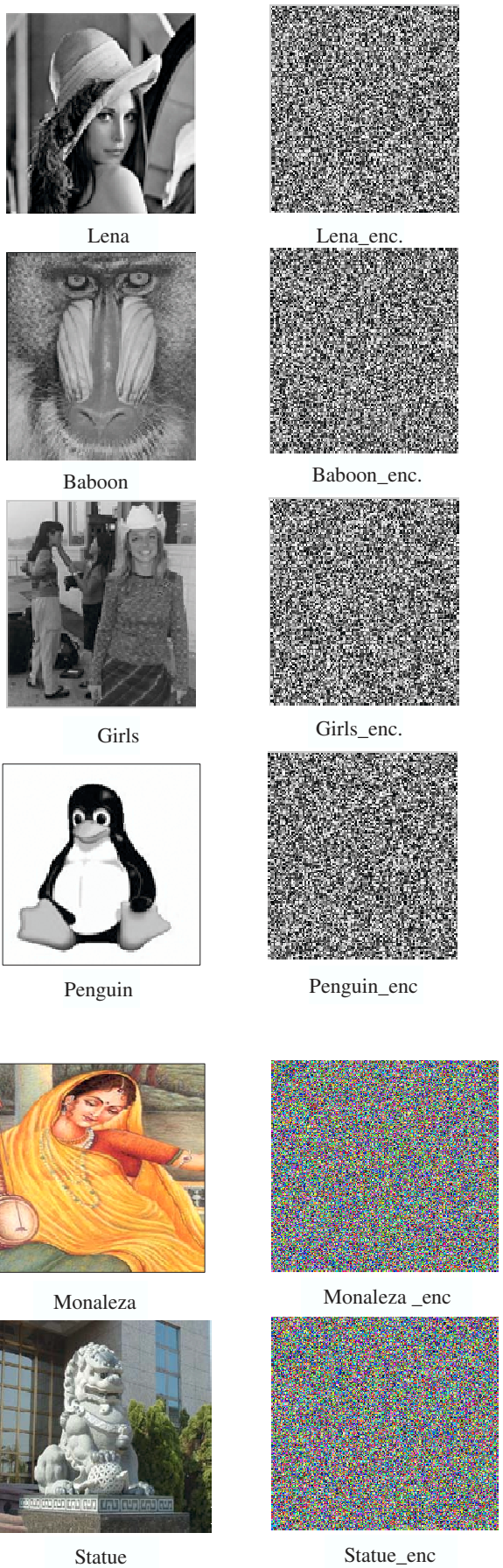

Figure 2: Encryption by the propose scheme

Table 1: The numerical evaluation for encryption quality

Image	Type	Correlation Coefficient		
		Chen [6]	Guan [10]	Proposed cipher
Lena	Gray	0.004200	0.008901	0.000044
Girls	Gray	0.006200	0.002700	0.000473
Penguin	Gray	0.011402	0.029700	0.001990
Penguin	Gray	0.006300	0.008005	0.000839
Airplane	Gray	0.002500	0.006700	0.000980
Nike	Gray	0.005800	0.005200	0.001600
Saturn	Gray	0.003900	0.013600	0.001960
Statue	color	0.004700	0.010400	0.000160
Penguin	color	0.009000	0.012600	0.006901
Monaleza	color	0.000530	0.006400	0.000853

[6, 16]. Some security analysis on the proposed image encryption scheme, including the most important ones like key space analysis, and statistical analysis, which demonstrated the satisfactory security of the proposed scheme, are described. Different images have been tested, and similar results are obtained. However, due to page limit, only the results for Lena (Figure 2) are used for illustration.

4.1 Key Space Analysis

A good image encryption algorithm should be sensitive to the cipher keys, and the key space should be large enough to make brute force attacks infeasible. For the proposed image encryption algorithm, key space analysis and testing have been performed and completely carried out, with results summarized as follows:

- *Key Space*
  The proposed image cipher has 2^{104} ($\approx 2.02824 \times 10^{31}$) different combinations of secret key. An image cipher with such a large key space is sufficient for reliable practical use and can resist all kinds of brute force attacks.

- *Key Sensitivity Test*
  An ideal image encryption procedure should be sensitive with respect to the secret key, i.e., the change of a single bit in the secret key should produce a completely different encrypted image. For testing the key sensitivity of the proposed encryption scheme, we have performed the following steps:

  1) An original images (Lena and Statue) are encrypted by using the secret key "32C69C29084AF8F23AF4AD0E99" (in hexadecimal) and the resultant image is referred as encrypted image A.
  2) The same original image is encrypted by making the slight modification in the secret key i.e. "33C69C29084AF8F23AF4AD0E99" (the most significant bit is changed in the secret key) and

the resultant image is referred as encrypted image B.

3) Again, the same original image is encrypted by making the slight modification in the secret key i.e. "32C69C29084AF8F23AF4AD0E98" (the least significant bit is changed in the secret key) and the resultant image is referred as encrypted image C.
4) Finally, the three encrypted images A, B and C are compared.

In Figure 3, we have shown the original image, the three encrypted images produced in the aforesaid steps, and the three difference images (A-B, A-C, and, B-C). It is not easy to compare the encrypted images by simply observing these images. So for comparison, we have calculated the correlation between the corresponding pixels of the three encrypted images. For this calculation, we have used the same formula as given in Equation (12) except that in this case and are the values of corresponding pixels in the two encrypted images to be compared. In Table 2, we have given the results of the correlation coefficients and pixel difference between the corresponding pixels of the three encrypted images A, B and C. It is clear from the table that no correlation exists among three encrypted images even though these have been produced by using slightly different secret keys.

Also, for example, The encrypted image by "32C69C29084AF8F23AF4AD0E99" has 99.59% of difference from the one encrypted by the key "32C69C29084AF8F23AF4AD0E98" in terms of pixel grey scale values, although there is only one bit difference in the two keys. Moreover, when a key is used to encrypt an image while another trivially modified (slightly different) key is used to decrypt the ciphered image, the decryption also completely fails. Figure4 shows that the image encrypted by "32C69C29084AF8F23AF4AD0E99" (image A) is not correctly decrypted by using the key "33C69C29084AF8F23AF4AD0E99" (image B) or by the key "32C69C29084AF8F23AF4AD0E98" (image C), although there is only one bit difference between the encryption and decryption keys. Hence the proposed encryption scheme is highly key sensitive.

4.2 Statistical Analysis

Statistical analysis on the proposed image encryption algorithm showed its superior confusion and diffusion properties which strongly resist statistical attacks. This is shown by a test on the histograms of the enciphered images and on the correlations of adjacent pixels in the ciphered image [6].

1) *Histograms of Encrypted Images*
Select several 256 grey-scale images of size $m \times n$ that have different contents, and calculate their histograms. Statistical analysis of Lena and Monaleza images and their encrypted images yielded their grey-scale histograms given in Figure 5. This figure shows

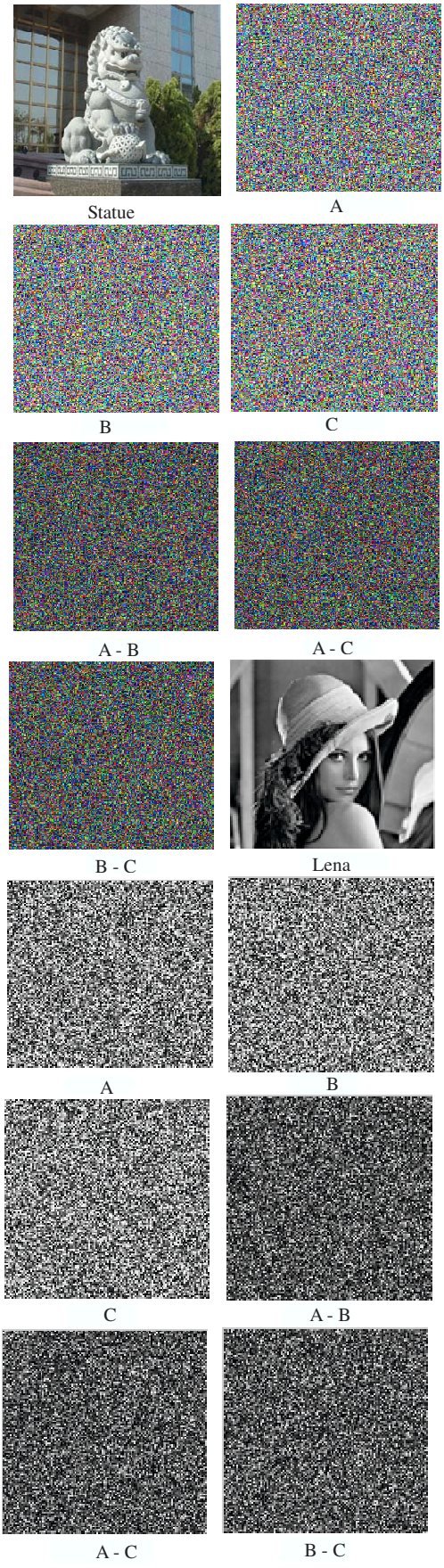

Figure 3: Key sensitivity test by encrypting the plain image by several slightly different keys

Table 2: Correlation coefficients and pixel difference between the corresponding pixels of the three different encrypted images obtained by using slightly different secret keys

Encrypted Image1	Encrypted Image2	Pixel differences		Coefficient Coefficient	
		Lena	Stature	Lena	Stature
Image A	Image B	99.56	99.53	0.005	0.0084
Image C	Image A	99.59	99.60	0.0016	0.0041
Image B	Image C	99.69	99.57	0.0005	0.0023

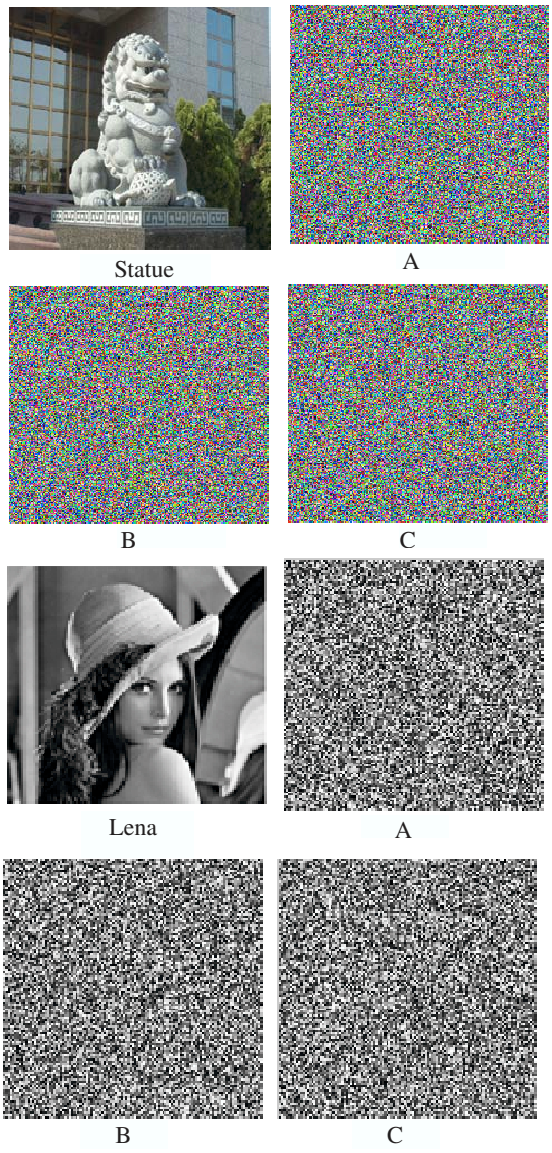

Figure 4: Key sensitivity test for decryption by using slightly different key from the encryption key

Table 3: Correlation coefficients of two adjacent pixels In original and encrypted images

Direction	Lena		Statue	
	Plain image	Cipher image	Plain image	Cipher image
Horizontal	0.9569	0.00061	0.9707	0.00940
Vertical	0.8919	0.00400	0.9671	0.00590
Diagonal	0.9223	0.0057	0.8231	0.00026

that the histogram of the ciphered images is fairly uniform and is significantly different from that of the original image. Also, it demonstrates that the encryption algorithm has covered up all the characters of the plain image and has complicated the dependence of the statistics of the output on the statistics of the input.

2) *Correlation of Two Adjacent Pixels*
To test the correlation between two adjacent pixels in plain-image and ciphered image, the following procedure was carried out. First, randomly select 1000 pairs of two adjacent (in horizontal, vertical, and diagonal direction) pixels from an image. Then, referring to [6], calculate the correlation coefficient of each pair by Equations (12). Figure 6 shows the correlation distribution of two horizontally adjacent pixels in the plain-image and that in the ciphered image. Similar results for diagonal and vertical directions were obtained, which are shown in Table 3. These correlation analysis prove that the proposed encryption technique satisfies zero co-correlation.

Also we note that, the statistic relation between secret encryption and cipher-image become complexity as possible, and attacker cannot educe the encryption key from cipher-image. Lena image and a full black image (namely, value of all pixels is 0) are encrypted using the encryption key "32C69C29084AF8F23AF4AD0E99". The histogram of plain-images and cipher-images are drawn in Figure 7, respectively. We find that, although the histogram of two plain-images has their distributive characters respectively, the histogram of two ciphered images are fairly uniform. It is very difficult to educe secret key from cipher-text when attacker try to attack by using the known-plaintext

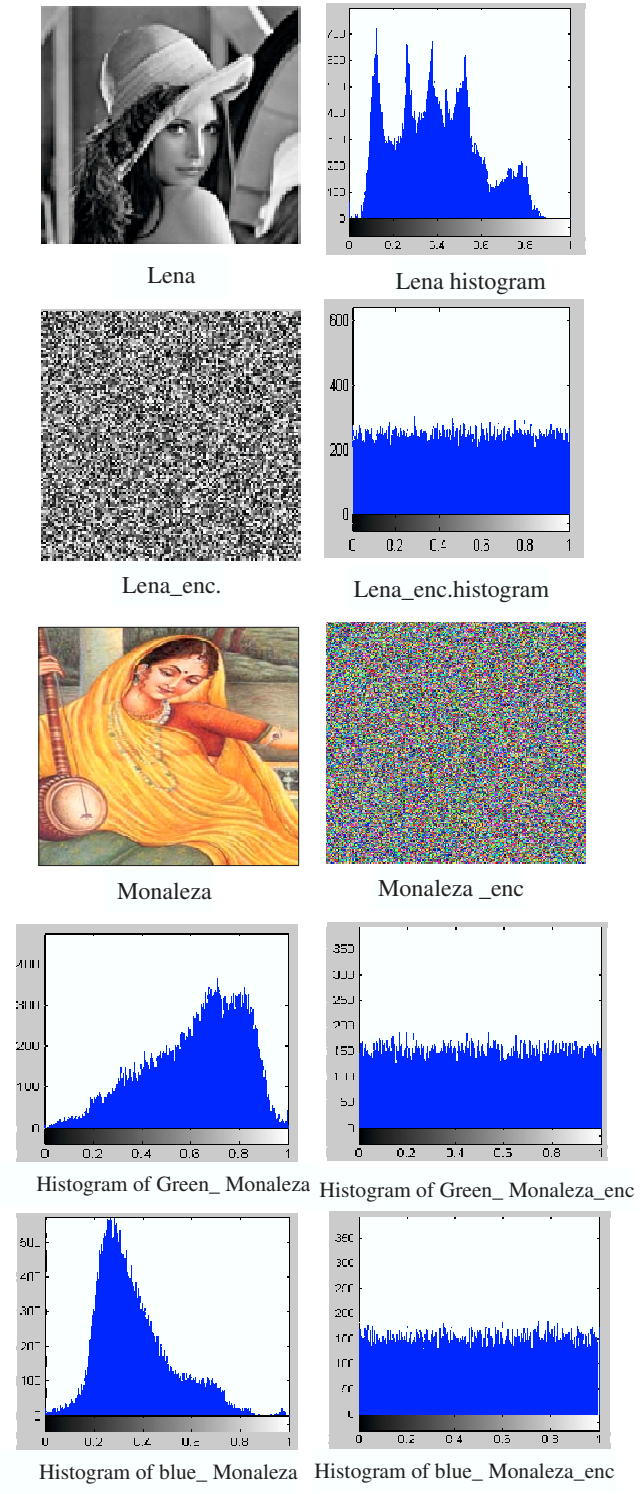

Figure 5: Histogram of the plain image and cipher image

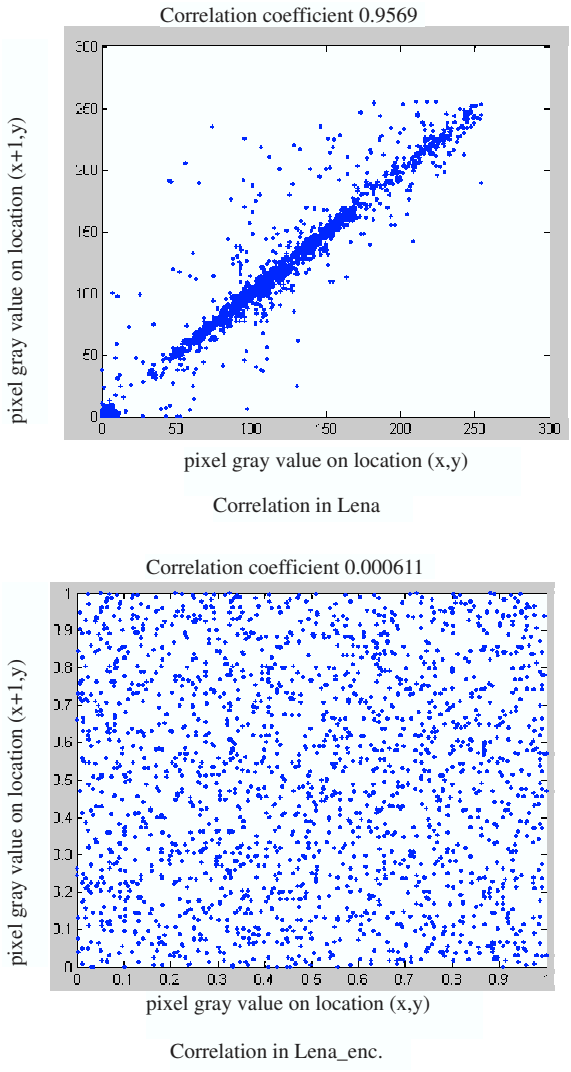

Figure 6: Correlation of two horizontally adjacent pixels in original and encrypted images.

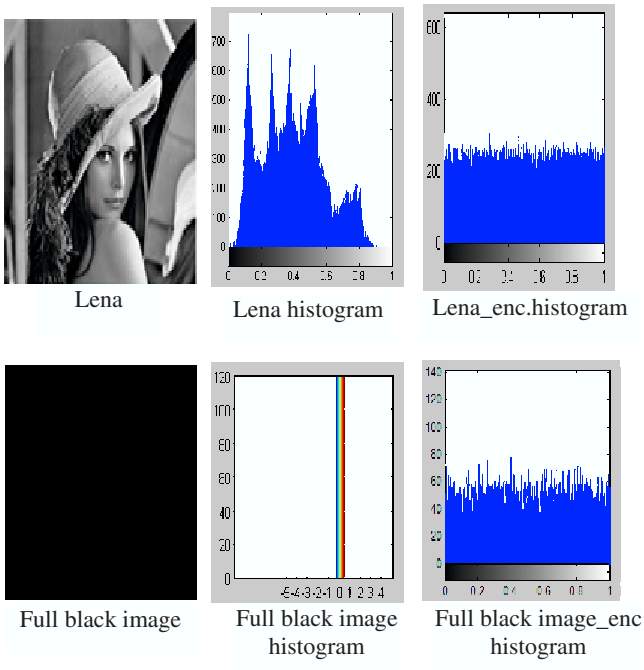

Figure 7: Histogram of the plain image and cipher image of Lena and full black image

attacks or chosen-plaintext attacks [2].

4.3 Differential Attacks

A desirable property for the proposed cipher is its sensitive to small change in the plain-image (single bit change in plain-image). To test the influence of one-pixel change on the plain-image, encrypted by the proposed cipher, two common measures may be used: Number of Pixels Change Rate *(NPCR)* and Unified Average Changing Intensity *(UACI)*. Consider two ciphered images, whose corresponding plain-images have only one pixel difference, be denoted by $C1$ and $C2$. Label the gray-scale values of the pixels at *grid* (i,j) in $C1$ and $C2$ by $C1(i,j)$ and $C2(i,j)$, respectively. Define a bipolar array, D, with the same size as images $C1$ and $C2$. Then, $D(i,j)$ is determined by $C1(i,j)$ and $C2(i,j)$, namely, if $C1(i,j) = C2(i,j)$ then $D(i,j) = 0$; otherwise, $D(i,j) = 1$. First, The $NPCR$ is defined as

$$NPCR = \frac{\sum_{i,j} D(i,j)}{W \times H} \times 100\%$$

Where W and H are the width and height of $C1$ or $C2$. The $NPCR$ measures the percentage of different pixel numbers between the plain-image and the cipher-image. Second, The $UACI$ is defined as

$$UACI = \frac{1}{W \times H}[\sum_{i,j} \frac{|C_1(i,j) - C_2(i,j)|}{255}] \times 100\%$$

which measures the average intensity of differences between the two images.

Some tests have been performed on the proposed scheme about the influence of only one-pixel change on the 256 grey scale image and color image. With respect to NPCR estimation, NPCR is obtained using the proposed cipher and estimated to be over 99.54% showing thereby that the encryption scheme is very sensitive with respect to small changes in the plain-image. With respect to UACI estimation, UACI is calculated to be 33.49% indicating that the rate of influence due to one pixel change is very large. Generally, these obtained results for NPCR and UACI show that the proposed algorithm is very sensitive with respect to plain-image (plain-images have only one pixel difference).

5 Conclusion

In this paper, some characters possessed by chaotic system are utilized to design a potential method for image encryption. In the proposed image encryption scheme, an external secret key of 104 bit and two chaotic logistic maps are employed to confuse the relationship between the cipher image and the plain image. Further, to make the cipher more robust against any attack, the secret key is modified after encrypting of each pixel of the plain image. The robustness of the proposed system is further reinforced by a feedback mechanism, which makes the encryption of each plain pixel depends on the key, the value of the previous cipher pixel and the output of the logistic map (data dependent property). The experimental results demonstrated that the proposed image encryption technique have several interesting features, such as a high level of security, large enough key space, and pixel distributing uniformity. From the analysis of these results, it is clear that the proposed algorithm has many characteristics of traditional cryptography, such as almost zero correlation and has been successfully applied to and tested for the image encryption. Although the algorithm presented in this paper has focused on image encryption, it is not just limited to this area and can be widely applied in the secure transmission of confidential information over the Internet.

References

[1] Z. E. Abid and W. Wang, "Countermeasures for hardware fault attack in multi-prime RSA cryptosystems," *International Journal of Network Security*, vol. 6, no. 2, pp. 190-200, 2008.

[2] M. S. Baptista, "Cryptography with chaos," *Physics Letters A*, vol. 240, pp. 50-54, 1998.

[3] N. Bourbakis, and C. Alexopoulos, "Picture data encryption using SCAN pattern," *Pattern Recognition*, vol. 25, pp. 567-581, 1992.

[4] H. K. L. Chang, and J. L. Liu, "A linear quad tree compression scheme for image encryption," *Signal Processing*, vol. 10, no. 4, pp. 279-290, 1997.

[5] C. C. Chang, M. S. Hwang, and T. S. Chen, "A new encryption algorithm for image cryptosystems,"

Journal of Systems and Software, vol. 58, pp. 83-91, 2001.

[6] G. Chen, Y. Mao, and C.K. Chui, "A symmetric image encryption based on 3D chaotic maps," *Chaos Solitons Fractals*, vol. 21, pp. 749-761, 2004.

[7] H. Cheng, and X. B. Li, "Partial encryption of compressed image and videos," *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 2439-2451, 2000.

[8] D. S. A. Elminaam, H. M. A. Kader, and M. M. Hadhoud, "Evaluating the performance of symmetric encryption algorithms," vol. 10, no. 3, pp. 213-219, 2010.

[9] N. E. Fishawy, and O. M. A. Zaid, "Quality of encryption measurement of bitmap images with RC6, MRC6, and Rijndael block cipher algorithms," *International Journal of Network Security*, vol. 5, no. 3, pp. 241-251, 2007.

[10] Z. H. Guan, F. Huang, and W. Guan, *Chaos-Based Image Encryption Algorithm*, *Physics Letter A*, vol. 346, pp. 153-157, 2005.

[11] F. Huang, "A new general transparency model for block-based watermarking method," *International Journal of Network Security*, vol. 7, no. 2, pp. 235-239, 2008.

[12] F. Jiri, "Symmetric ciphers based on two dimensional chaotic maps," *International Journal of Bifurcat Chaos*, vol. 8, no. 6, pp. 1259-1284, 1998.

[13] J. N. Jr, "Analysis of Venkaiah et al.'s AES design," *International Journal of Network Security*, vol. 9, no. 3, pp. 285-289, 2009.

[14] S. Li, and X. Zheng, "Cryptanalysis of a chaotic image encryption method," *Proceedings of the IEEE International. symposium on circuits and systems*, pp. 708-711, Scottsdale, AZ, USA, 2002.

[15] S. Li, X. Zheng, X. Mou, and Y. Cai, "Chaotic encryption scheme for real time digital video," *Proceedings of the SPIE on electronic imaging*, pp. 149-160, San Jose, CA, USA, 2002.

[16] N. K. Pareek, V. Patidar, and K. K. Sud, *Image Encryption Using Chaotic Logistic Map*, *Image and Vision Computing*, vol. 24, pp. 926-934, 2006.

[17] P. Refregier, and B. Javidi, "Optical image encryption based on input plane and fourier plane random encoding," *Optics Letters*, vol. 20, pp. 767-769, 1995.

[18] J. Scharinger, "Fast encryption of image data using chaotic Kolmogrov flow," *Electronic Imaging*, vol. 7, no. 2, pp. 318-325, 1998.

[19] K. Wang, W. Pei, L. Zou, A. Song, and Z. He, "On the security of 3D cat map based symmetric image encryption scheme," *Physics Letter A*, vol. 343, pp. 432-439, 2005.

[20] J. C. Yen, and J. I. Guo, "A new image encryption algorithm and its VLSI architecture," *Proceedings of the IEEE workshop Signal Processing Systems*, pp. 430-437, 1999.

[21] J. C. Yen, and J. I. Guo, "An efficient hierarchical chaotic image encryption algorithm and its VLSI realization," *IEE Proceedings-Vision Image Processing*, vol. 147, pp. 167-175, 2000.

[22] J. C. Yen, and J. I. Guo, "A new chaotic key based design for image encryption and decryption," *Proceedings of the IEEE International Symposium Circuits and Systems*, vol. 4, pp. 49-52, 2000.

[23] X. Y. Zhao, and G. Chen, "Ergodic matrix in image encryption," *Procssdings Second International Conference on Image and Graphics*, vol. 4875, pp. 394-401, 2002.

Ismail Amr Ismail is Professor, Dean, College of Computers and informatics, Misr International University, Egypt 2007-up to date, Dean, college of computers and informatics, Zagazig University, 2000-2006. He is assistant, associate, professor of computational math, college of science zagazig University, 1980-2000. His research interests include pattern analysis and machine intelligence, data structures and analysis, genetic algorithms, neural network, image processing, cryptography, parallel computing, and database.

Mohammed Amin was graduated in mathematics in 1983 at Menoufia University. He studied computer science from 1986 to 1989 at Ain Shams University in Cairo and received the M.Sc. degree in 1990 and the Ph.D degree in computer science in 1997 at the University of Gdansk, Poland. He is associate professor of computer science at the faculty of science, Menoufia University, and research visitor to the faculty of Philosophy and sciences of the Silesian University, Opava, Czech Republic. His research area in formal languages and their application in compiler design. Cooperating/distributed systems, web information retrieval, Petri nets and its applications, and finite automata and cryptograph.

Hossam Diab was graduated in computer science in 1999 and received the M.Sc. degree in 2004 at the faculty of science, Menoufia University, Egypt. He is assistant lecturer at the department of mathematics and computer science, faculty of science, Menoufia University, Egypt. He is working for his Ph.D. His research interests are in the areas of digital image processing, image compression, security over wired and wireless networks, image watermarking and networking protocols.