An Efficient Chaos-Based Feedback Stream Cipher (ECBFSC) for Image Encryption and Decryption

Hossam El-din H. Ahmed, Hamdy M. Kalash, and Osama S. Farag Allah
Menoufia University, Faculty of Electronic Engineering, Dept. of Computer Science & Engineering,
32952, Menouf, Egypt
E-mail of corresponding author: osam_sal@yahoo.com

Keywords: stream cipher, chaos, logistic map, security analysis

Received: September 23, 2005

The chaos based cryptographic algorithms have suggested some new and efficient ways to develop secure image encryption techniques. Towards this direction, this paper presents an efficient chaos-based feedback stream cipher (ECBFSC) for image cryptosystems. The proposed stream cipher is based on the use of a chaotic logistic map and an external secret key of 256-bit. The initial conditions for the chaotic logistic map are derived using the external secret key by providing weightage to its bits corresponding to their position in the key. Further, new features of the proposed stream cipher include the heavy use of data-dependent iterations, data-dependent inputs, and the inclusion of three independent feedback mechanisms. These proposed features are verified to provide high security level. A complete specification for the proposed ECBFSC is given. Several test images are used for inspecting the validity of the proposed ECBFSC. The results of several experimental, key space analysis, statistical analysis, and key sensitivity tests show that the proposed ECBFSC for image cryptosystems provides an efficient and secure way for real-time image encryption and transmission from the cryptographic viewpoint.

Povzetek: Opisan je na teoriji kaosa razvit kriptografski algoritem.

1 Introduction

During the last decade, the use of computer networks has grown spectacularly, and this growth continues unabated. Almost all networks are being installed, interconnected, and connected to the global internet. Today more and more information has been transmitted over the internet. The information is not only text, but also audio, image, and other multimedia. Images have been widely used in our daily life. However, the more extensively we use the images, the more important their security will be. For example, it is important to protect the diagrams of army emplacements, the diagrams of bank building construction, and the important data captured by military satellites. In addition, the number of computer crimes has increased recently. Image security has become an important topic in the current computer world.

Many encryption methods have been proposed in literature, and the most common way to protect large multimedia files is by using conventional encryption techniques. Implementations of popular public key encryption methods, such as RSA or El-Gamal [1] cannot provide suitable encryption rates, while security of these algorithms relies on the difficulty of quickly factorizing large numbers or solving the discrete logarithm problem, topics that are seriously challenged by recent advances in number theory and distributed computing.

On the other hand, private key bulk encryption algorithms, such as Triple DES or Blowfish [2], are more suitable for transmission of large amounts of data. However, due to the complexity of their internal structure, they are not particularly fast in terms of execution speed and cannot be concisely and clearly explained, so that to enable detection of cryptanalytic vulnerabilities.

Chaotic maps present many desired cryptographic qualities such as simplicity of implementation that leads to high encryption rates, and excellent security.

In this paper we propose an efficient chaos-based feedback stream cipher (ECBFSC) for image cryptosystems. The proposed ECBFSC works using an iterative cipher mechanism that is based on the logistic function. The encryption module encrypts the image pixel-by-pixel, taking into consideration, in each iteration, the values of the previously encrypted pixels. This feedback property, combined with the external secret key of 256-bit, makes our stream cipher robust against cryptanalytic attacks. Furthermore a simple implementation of ECBFSC achieves high encryption rates on general-purpose computers.

The rest of this paper is organized as follows: Section 2 surveys some related image cryptosystems. We discuss some characteristics of an image cryptosystem and research issues of an image cryptosystem in Sections 3 and 4 respectively. Section 5 presents chaos and cryptography including characteristics and analysis of

chaotic logistic map. Section 6 examines the step by step procedure of encryption/decryption modules for the proposed ECBFSC. Section 7 explores design Principles of the ECBFSC. Test, verification and efficiency of the proposed ECBFSC are given in Section 8. Section 9 discusses the detailed security analysis of the proposed ECBFSC including key space analysis, statistical analysis, and sensitivity analysis with respect to key and plaintext. Performance evaluation of the proposed ECBFSC is explored in Section 10. Finally, Section 11 concludes this paper.

2 Related Image Cryptosystems

2.1 Picture data encryption using SCAN patterns

First sub Bourbakis and Alexopoulos [3] developed another method to encrypt images. This method converts a 2D image into a 1D list, and employs a SCAN language [4] to describe the converted result. In this language, there are several SCAN letters. Each SCAN letter represents one kind of scan order. Different kinds of combinations of SCAN letters may generate different kinds of secret images. After determining the combination of SCAN letters, the scheme then generates a SCAN string. This string defines the scan order of the original image. Next, this method scans the original image in the determined order and, moreover, encrypts the SCAN string by using commercial cryptosystems. Since the illegal users cannot obtain the correct SCAN string, the original image is therefore secure. There is no image compression in this method. Therefore, the size of the image is very large, and thus it is inefficient to encrypt or decrypt the image directly.

2.2 Novel image encryption technique and its application in progressive transmission

Kuo proposed an encryption method that referred to the image distortion [5]. This method obtains the encrypted image by adding the phase spectra of the plainimage with those of another key image. Since the phase spectra of the original image are randomly changed, the cipherimage is unrecognizable. Thus this method is safe, but no image compression is considered.

2.3 An image encryption scheme based on quadtree compression scheme

Chang and Liou [6] proposed an encryption method for images. This method employs two technologies to achieve the compression and encryption purposes. They are the quadtree data structure and the SCAN language, respectively. This method first compresses the original image by using a quadtree, and then encrypts the compressed data by SCAN. So, this method can compress and encrypt images concurrently. Quadtree is notably a lossless data compression technology. Therefore, this method is also lossless.

2.4 A New Encryption Algorithm for Image Cryptosystems

Chin-Chen Chang, Min-Shian Hwang, and Tung-Shou Chen [7] use one of the popular image compression techniques, vector quantization to design an efficient cryptosystem for images. The scheme is based on vector quantization (VQ), cryptography, and other number theorems. The images are first decomposed into vectors and then sequentially encoded vector by vector.

2.5 Symmetric ciphers based on two dimensional chaotic maps

Fridrich [8] demonstrated the construction of a symmetric block encryption technique based on two-dimensional standard baker map. There are three basic steps in the method of Fridrich [8]: (a) choose a chaotic map and generalize it by introducing some parameter, (b) discretize the chaotic map to a finite square lattice of points that represent pixels, (c) extend the discretized map to three-dimensions and further compose it with a simple diffusion mechanism. Further,

2.6 Fast encryption of image data using chaotic Kolmogrov

Scharinger [9] designed a chaotic Kolmogrov-flow-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.

2.7 A new image encryption algorithm and its VLSI architecture

Yen and Guo [10] 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.

2.8 A new chaotic key based design for image encryption and decryption

Further, Yen and Guo [11] 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.

2.9 Chaotic encryption scheme for real time digital video

Recently, Li et al. [12] have proposed a video encryption technique based on multiple digital chaotic systems which is known as CVES (Chaotic Video Encryption Scheme). In this scheme, 2n chaotic maps are used to generate pseudo random signals to mask the video and to perform pseudo random permutation of the masked video. Very recently,

2.10 A symmetric image encryption based on 3D chaotic maps

Chen et al. [13] 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.section text.

3 Characteristics of An Image Cryptosystem

A good information security system is able to not only protect confidential messages in the text form, but also in image form. In general, there are three basic characteristics in the information security field: privacy, integrity, and availability [14].

1. Privacy: an unauthorized user cannot disclose a message.
2. Integrity: an unauthorized user cannot modify or corrupt a message.
3. Availability: messages are made available to authorized users faithfully.

A perfect image cryptosystem is not only flexible in the security mechanism, but also has high overall performance. Thus, besides the above characteristics, the image security also requires the following characteristics:

1. The encryption system should be computationally secure. It must require an extremely long computation time to break, for example. Unauthorized users should not be able to read privileged images.
2. Encryption and decryption should be fast enough not to degrade system performance. The algorithms for encryption and decryption must be simple enough to be done by users with a personal computer.
3. The security mechanism should be as widespread as possible. It must be widely acceptable to design a cryptosystem like a commercial product.
4. The security mechanism should be flexible.
5. There should not be a large expansion of the encrypted image data.

4 Research issues of An Image Cryptosystem

According to the analyses stated in Section 2, there are forth research issues on image cryptosystems as follows:

The first issue is to encrypt the image data using the same method as for text data. Images are usually represented as 2D arrays. They should be converted into 1D arrays before enciphering. Various encryption techniques can be used and applied on the 1D lists such as in [3-5]. Since the image is large, it is inefficient to encrypt or decrypt the picture directly. Applying compression techniques to images and then encrypting the compressed images is also a way to use standard text encryption algorithms.

The second issue is to use the special features of images. The main feature of an image is that it allows a bit of distortion. Therefore, picture data can be compressed before transmitting, and be lossy decompressed with a small distortion after receiving the image compression. There are many lossy compression techniques for images. This issue is to encrypt the compressed image using the same method as for the text data. Since the size of the compressed image is usually larger than that of text data, it is also invalid to reduce the size of the picture by image compression before enciphering. Chang and Liou's image cryptosystem [6] is in this form.

The third issue depend on the use of vector quantization (VQ), cryptography, number theorems and compression. The auxiliary data is encrypted only by some encryption algorithms such as in [7]. Since the size of the auxiliary data is usually less than that of the compressed image, the time complexity for enciphering auxiliary data is less than that of the above two issues.

The forth issue depends on chaotic maps that are considered as candidate for design of chaos based encryption techniques [8-13] which are good for practical use as these techniques provide a good combination of speed, high security, complexity, reasonable computational overheads and computational power etc. The proposed ECBFSC belongs to this issue.

5 Chaos and Cryptography

The close relationship between chaos and cryptography makes chaos based cryptographic algorithms as a natural candidate for secure communication and cryptography

chaos based encryption techniques are considered good for practical use as these techniques provide a good combination of speed, high security, complexity, reasonable computational overheads and computational power etc.

5.1 Characteristics of the chaotic maps

The characteristics of the chaotic maps have attracted the attention of cryptographers since it has many

fundamental properties such as ergodicity, sensitivity to initial condition and system parameter, and mixing property, etc [15-16].
Most properties are related to some requirements such as mixing and diffusion in the sense of cryptography. Therefore, chaotic cryptosystems have more useful and practical applications.

5.2 The logistic map and its analysis

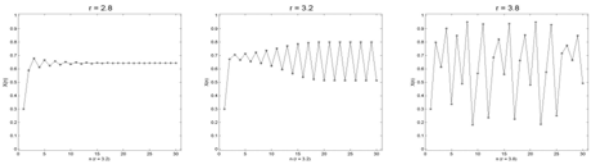
One of the simplest chaos functions that have been studied recently for cryptography applications is the logistic map. The logistic map function is expressed as:

$$X_{n+1} = rX_n(1-X_n) \quad (1)$$

Where X_n takes values in the interval [0,1]. It is one of the simplest models that present chaotic behavior [17].

The parameter r can be divided into three segments, which can be examined by experiments on following conditions: $X_0 = 0.3$. When $r \in [0,3]$ as shown in Fig. 1(a), the calculation results come to the same value after several iterations without any chaotic behaviour. When $r \in [3,3.57]$, the phase space concludes several points only, as showed in Fig. 1(b), the system appears periodicity. While $r \in [3.57,4]$, it becomes a chaotic system with periodicity disappeared as shown in Fig. 1(c). So we can draw the following conclusions:

(1) When $r \in [0,3.57]$, the points concentrate on several values and could not be used for image cryptosystem.
(2) For $r \in [3.57,4]$, the logistic map exhibits chaotic behavior, and hence the property of sensitive dependence [18]. So it can be used for image cryptosystem.



a) iteration property when r = 2.8 b) iteration property when r = 3.2 c) iteration property when r = 3.8

Figure 1: Analysis of Logistic Map

6 The Proposed ECBFSC

In this section, we discuss the step by step procedure of the proposed ECBFSC for encryption as well as decryption process. The proposed ECBFSC consists of two main modules, the encryption and decryption module.

6.1 The encryption module

An overview of ECBFSC encryption module is depicted in Fig. 2. the proposed ECBFSC is a simple block cipher with block size of 8-bit and 256-bit secret key. The key is used to generate a pad that is then merged with the plaintext a byte at a time.

1. For the encryption/decryption, we divide plaintext/ciphertext into blocks of 8-bits. Plaintext and ciphertext of i blocks can be represented as

$$P = P_1P_2P_3P_4.....P_i \quad (2)$$

$$C = C_1C_2C_3C_4.....C_i \quad (3)$$

2. The proposed image encryption process utilizes an external secret key of 256-bit long. Further, the secret key is divided into blocks of 8-bit each, referred as session keys.

$$K = K_1K_2K_3K_4.....K_{64} \text{ (in hexadecimal)} \quad (4)$$

here, K_i's are the alphanumeric characters (0–9 and A–F) and each group of two alphanumeric characters represents a session key. Alternatively, the secret key can be represented in ASCII mode as

$$K = K_1K_2K_3K_4.....K_{32} \text{ (in ASCII)} \quad (5)$$

here, each K_i represents one 8-bit block of the secret key i.e. session key.

3. The initial condition (X_0) for the chaotic map and the initial code C_0 are generated from the session keys as

$$R = \sum_{i=1}^{32}(M1[K_i]) \quad (6)$$

$$X_0 = R - \lfloor R \rfloor \quad (7)$$

$$C_0 = \left[\sum_{i=1}^{32}(K_i)\right] \bmod 256 \quad (8)$$

here K_i, $\lfloor\ \rfloor$, and M1 are, respectively, the decimal equivalent of the ith session key, the floor function, and mapping from the session, key space, all integers between 0 and 255, into the domain of the logistic map, all real numbers in the interval [0,1].

4. Read a byte from the image file (that represent a block of 8-bits) and load it as plainimage pixel P_i.
5. Encryption of each plainimage pixel P_i to produce its corresponding cipherimage pixel C_i can be expressed mathematically as:

$$C_i = \left(P_i + M2\left[\sum_{i=1}^{\#_i} rX_i(1-X_i)\right]\right) \bmod 256 \quad (9)$$

Where X_i represents the current input for logistic map and computed as:

$$X_i = M1[X_{i-1} + C_{i-1} + K_i] \quad (10)$$

$\#_i$ is the number of iteration of logistic map for its current input X_i and calculated as:

$$\#_i = K_{i+1} + C_{i-1} \quad (11)$$

And M2 maps the domain of the logistic map, [0,1], back into the interval [0,255].

6. Repeat steps 4-5 until the entire image file is exhausted.

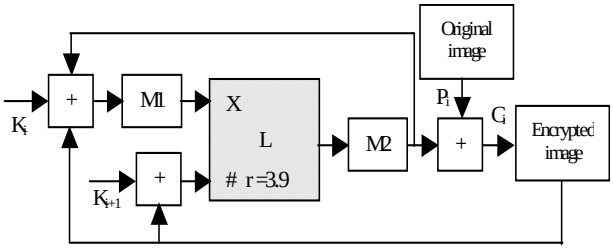


Figure 2: Diagram of Encryption Module

6.2 The decryption module

Decryption is very simple, the same pad is generated but this time un-merged with the ciphertext to retrieve the plaintext.

The diagram of ECBFSC decryption module is given in Fig. 3. The decryption module receives an encrypted image (cipherimage) and the 256-bit secret key and returns the original image (plainimage).

In particular, the decryption module works in the same way as the encryption module but now the output of the logistic map is subtracted from the corresponding cipherimage pixel C_i providing the plainimage pixel P_i . The output of the decryption module is the original image (plainimage).

Decryption of each cipherimage pixel C_i to produce its corresponding plainimage pixel P_i can be expressed mathematically as:

$$P_i = \left(C_i - M2\left[\sum_{i=1}^{\#_i} rX_i(1-X_i) \right] \right) \bmod 256 \qquad (12)$$

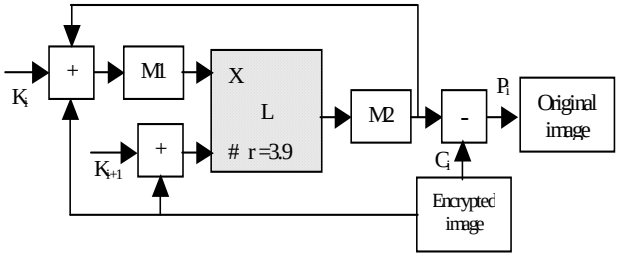


Figure 3: Diagram of Decryption Module

7 Design Principles of the Proposed ECBFSC

The basic concept is that the encryption of each part of the plainimage depends not only on the key, but also on the previous cipherimage.

The use of feedback mechanism has two desirable benefits. The first benefit is that there can be no simple periodicity in the encrypted image (cipherimage) because the encryption of each plainimage pixel depends not only on the encryption key, but also on the previous cipherimage pixel. The second benefit is that any changes in the plainimage are cascaded forward throughout the cipherimage, which means that two almost identical plainimages will encrypt to completely different cipherimages. This sensitivity to the plainimage is also a plus to the security of the proposed ECBFSC.

The proposed ECBFSC makes heavy use of data-dependent essentials. This appears for the current input of logistic map, which is data-dependent since it is computed as a function of the current session key K_i, previous computed cipher pixel C_{i-1} and previous logistic output. Also, the number of iterations # for the chaotic logistic map is data-dependent since it is computed as a function of current session key K_{i+1} and previous computed cipher pixel C_{i-1}.

As we encrypt each new block, i, the counter used to keep track of the current session key, is incremented. The output of the logistic map is then merged with the plaintext to give the ciphertext.

8 Test, Verification and Efficiency of ECBFSC

Results of some experiments are given to prove its efficiency of application to digital images.

We use the gray-scale images--Lena and Eiffel Tower, each of size 256 x 256, gray-scale (0-255) as the original images (plainimages) and the secret key "12345789012345678912345678901 2" (in ASCII) is used for encryption whose size is 256-bit. The encrypted images are depicted in Figs. 4(b)-5(b). As shown, the encrypted images (cipherimages) regions are totally invisible.

The decryption method takes as input the encrypted image (cipherimage), together with the same secret key "12345789012345678912345678901 23" (in ASCII). The decrypted images are shown in Figs. 4(c)-5(c).

The visual inspection of Figs. (4-5) shows the possibility of applying the proposed ECBFSC successfully in both encryption and decryption. Also, it reveals its effectiveness in hiding the information contained in them.

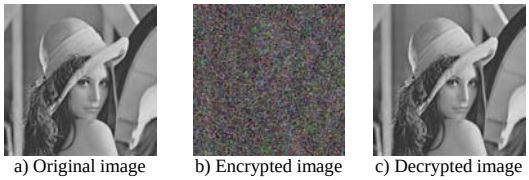

a) Original image b) Encrypted image c) Decrypted image

Figure 4: Application of ECBFSC to Lena Plainimage/Cipherimage


a) Original image b) Encrypted image c) Decrypted image

Figure 5: Application of ECBFSC to Eiffel Tower Plainimage/Cipherimage

9 Security Analysis and Test Results

A good encryption procedure should be robust against all kinds of cryptanalytic, statistical and brute-force attacks. In this section, we discuss the security

analysis of the proposed ECBFSC such as key space analysis, statistical analysis, and sensitivity analysis with respect to the key and plainimage to prove that the proposed cryptosystem is secure against the most common attacks [19-22].

9.1 Key space analysis

For a secure image cryptosystem, the key space should be large enough to make the brute force attack infeasible. The proposed ECBFSC has 2^{256} different combinations of the secret key. An image cipher with such a long key space is sufficient for reliable practical use.

In the proposed ECBFSC, a chaotic logistic map is employed which is sensitive on the initial condition. The initial condition for logistic map is calculated from the secret key.

Additionally the number of iterations supported by the logistic map module is between 0 and 767, as cipher pixels take values in the interval [0,512] and the session keys take values in the interval [0,255].

9.2 Statistical analysis

It is well known that many ciphers have been successfully analyzed with the help of statistical analysis and several statistical attacks have been devised on them. Therefore, an ideal cipher should be robust against any statistical attack. To prove the robustness of the proposed ECBFSC, we have performed statistical analysis by calculating the histograms and the correlations of two adjacent pixels in the plainimage/cipherimage.

9.2.1 Histograms analysis

To prevent the leakage of information to an opponent, it is also advantageous if the cipherimage bears little or no statistical similarity to the plainimage. An image-histogram illustrates how pixels in an image are distributed by graphing the number of pixels at each color intensity level. We have calculated and analyzed the histograms of the several encrypted images as well as its original images that have widely different content.

One typical example among them is shown in Fig. 6(b). The histogram of a plainimage contains large spikes. These spikes correspond to color values that appear more often in the plainimage.

The histogram of the cipherimage as shown in Fig. 6(d), is more uniform, significantly different from that of the original image, and bears no statistical resemblance to the plainimage. It is clear that the histogram of the encrypted image is fairly uniform and significantly different from the respective histograms of the original image and hence does not provide any clue to employ any statistical attack on the proposed image encryption procedure.

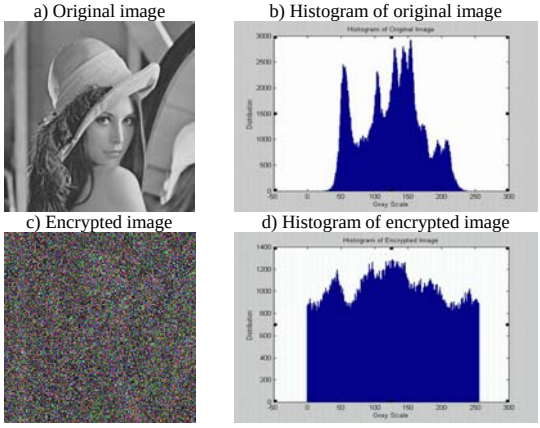


Figure 6: Histograms of the plainimage and the cipherimage

9.2.2 Correlation coefficient analysis

In addition to the histogram analysis, we have also analyzed the correlation between two vertically adjacent pixels, two horizontally adjacent pixels and two diagonally adjacent pixels in plainimage/cipherimage respectively. The procedure is as follows: First, randomly select 1000 pairs of two adjacent pixels from an image. Then, calculate their correlation coefficient using the following two formulas:

$$\mathrm{cov}(x,y) = E(x - E(x))(y - E(y)), \tag{13}$$

$$r_{xy} = \frac{\mathrm{cov}(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}, \tag{14}$$

Where x and y are the values of two adjacent pixels in the image. In numerical computations, the following discrete formulas were used:

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{15}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N} (x_i - E(x))^2, \tag{16}$$

$$\mathrm{cov}(x,y) = \frac{1}{N}\sum_{i=1}^{N} (x_i - E(x))(y_i - E(y)), \tag{17}$$

Fig. 7 shows the correlation distribution of two horizontally adjacent pixels in plainimage/cipherimage for the proposed ECBFSC. The correlation coefficients are 0.9905 and 0.0308 respectively for both plainimage/cipherimage. Similar results for diagonal and vertical directions are obtained as shown in Table 1. It is clear from the Fig. 7 and Table 1 that there is negligible correlation between the two adjacent pixels in the cipherimage. However, the two adjacent pixels in the plainimage are highly correlated.

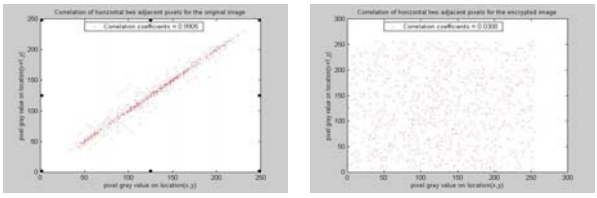


Figure 7: Two horizontally adjacent pixels Correlation in plainimage/cipherimage

Direction of Adjacent pixels	Plainimage	Cipherimage
Horizontal	0.9905	0.0308
Vertical	0.9787	0.0304
Diagonal	0.9695	0.0317

Table 1: Correlation coefficients in plainimage/ cipherimage

9.3 Sensitivity analysis

An ideal image encryption procedure should be sensitive with respect to both the secret key and plainimage. The change of a single bit in either the secret key or plainimage should produce a completely different encrypted image. To prove the robustness of the proposed ECBFSC, we will perform sensitivity analysis with respect to both key and plainimage.

9.3.1 Key sensitivity analysis

High key sensitivity is required by secure image cryptosystems, which means that the cipherimage cannot be decrypted correctly although there is only a slight difference between encryption or decryption keys. This guarantees the security of the proposed ECBFSC against brute-force attacks to some extent.

For testing the key sensitivity of the proposed image encryption procedure, we have performed the following steps:

(a) An original image in Fig. 8(a) is encrypted by using the secret key "12345789012345678912345678901 2" (in ASCII) and the resultant image is referred as encrypted image A as shown in Fig. 8(b).

(b) The same original image is encrypted by making the slight modification in the secret key i.e. "22345789012345678912345678901 2" (in ASCII) (the most significant bit is changed in the secret key) and the resultant image is referred as encrypted image B as shown in Fig. 8(c).

(c) Again, the same original image is encrypted by making the slight modification in the secret key i.e. secret key "12345789012345678912345678901 3" (in ASCII) (the least significant bit is changed in the secret key) and the resultant image is referred as encrypted image C as shown in Fig. 8(d).

(d) Finally, the three encrypted images A, B and C are compared.

In Fig. 8, we have shown the original image as well as the three encrypted images produced in the aforesaid steps. 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 Eq. (14) except that in this case x and y 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 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.

Key sensitivity analysis shows that changing one bit in encryption key will result in a completely different cipherimage by more than 99% in terms of pixel gray scale values.

Moreover, in Fig. 9, we have shown the results of some attempts to decrypt an encrypted image with slightly different secret keys than the one used for the encryption of the original image. Particularly, in Fig. 9(a) and Fig. 9(b) respectively, the original image and the encrypted image produced using the secret key "12345789012345678912345678901 2" (in ASCII) are shown whereas in Fig. 9(c) and Fig. 9(d) respectively, the images after the decryption of the encrypted image (shown in Fig. 9(b)) with the secret keys "12345789012345678912345678901 2" (in ASCII) and "12345789012345678912345678901 1" (in ASCII). It is clear that the decryption with a slightly different key fails completely and hence the proposed image encryption procedure is highly key sensitive.

Figure 8: Key sensitive test result 1 with ECBFSC

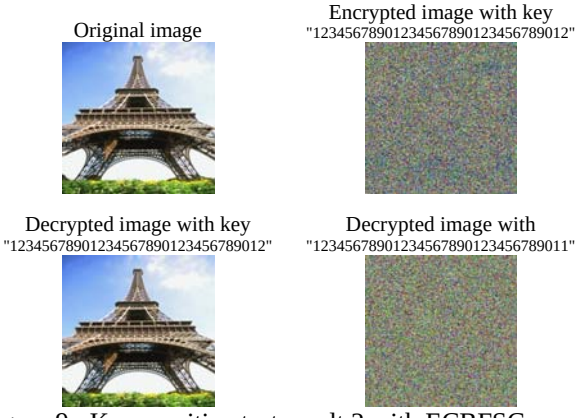


Figure 9: Key sensitive test result 2 with ECBFSC

Image 1	Image 2	Correlation coefficient
Encrypted image A Fig. 8(b)	Encrypted image B Fig. 8(c)	0.0326
Encrypted image B Fig. 8(c)	Encrypted image C Fig. 8(d)	0.0370
Encrypted image C Fig. 8(d)	Encrypted image A Fig. 8(b)	0.0369

Table 2: Correlation coefficients between the corresponding pixels of the three different encrypted images obtained by using slightly different secret key of an image shown in Fig. 8.

9.3.2 Plainimage sensitivity analysis

A desirable property for the proposed ECBFSC is that it is highly sensitive to small change in the plainimage (single bit change in plainimage).

To test the influence of one-pixel change on the plainimage, encrypted by the proposed ECBFSC, two common measures may be used: Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI). Let two ciphered images, whose corresponding plainimages 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) = 1; otherwise, D(i,j) = 0.

The NPCR is defined as

$$NPCR = \frac{\sum_{i,j} D(i,j)}{W \times H} \times 100\%, \qquad (18)$$

Where W and H are the width and height of C1 or C2. The NPCR measures the percentage of different pixel numbers between plainimage and cipherimage.

The UACI is defined as

$$UACI = \frac{1}{W \times H}\left[\sum_{i,j} \frac{C_1(i,j) - C_2(i,j)}{255}\right] \times 100\% \qquad (19)$$

Which measures the average intensity of differences between the two images. One performed test is on the one-pixel change influence on a 256 grey-level Lena image of size 256 x 256.

With respect to NPCR estimation. NPCR is obtained using the proposed ECBFSC and estimated to be over 99.62% showing thereby that the encryption scheme is very sensitive with respect to small changes in the plainimage.

With respect to UACI estimation. UACI is calculated to be %80.51 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 ECBFSC is very sensitive with respect plainimage (plainimages have only one pixel difference).

10 Performance Evaluation

Apart from the security consideration, some other issues on image encryption are also important. These include the running speed for real-time image encryption/decryption. The simulator for the proposed ECBFSC is implemented using the compiler in Borland C++ Development Suite 5.0. Performance was measured on a 2.4 GHz Pentium IV with 256 Mbytes of RAM running Windows XP. In addition, to improve the accuracy of our timing measurements, each set of the timing tests shown in Table 3 was executed 10 times, and we report the average of the times thereby obtained.

Image size (in pixels)	Colors	Encryption in Sec.	Decryption in Sec.
256 x 256	2	< 0.0010	< 0.0010
256 x 256	16	< 0.0010	< 0.0010
256 x 256	256	0.0030	0.0040
256 x 256	16777216	0.0267	0.0360
512 x 512	2	< 0.0010	< 0.0010
512 x 512	16	0.0090	0.0108
512 x 512	256	0.0305	0.0358
512 x 512	16777216	0.1108	0.1306
1024 x 1024	2	0.0150	0.0163
1024 x 1024	16	0.0716	0.0832
1024 x 1024	256	0.1618	0.1744
1024 x 1024	16777216	0.4690	0.587
2048 x 2048	2	0.0666	0.0954
2048 x 2048	16	0.2942	0.3802
2048 x 2048	256	0.6322	0.7604
2048 x 2048	16777216	1.6085	1.8100

Table 3: Enciphering/deciphering speed test results of the proposed ECBFSC

Table 3 summarizes the encryption/decryption speeds for the proposed ECBFSC on images of different sizes. The results emphasis that proposed ECBFSC. Simulation results show that the average encryption/decryption speed is 7.46 MB/Sec for encryption and 6.63 MB/Sec for decryption. The peak speed can reach up to 7.6 MB/Sec for encryption and 6.7 MB/Sec for decryption.

11 Conclusion

In this paper, a new way of image encryption scheme have been proposed which utilizes a chaos-based feedback cryptographic scheme using the logistic map and an external secret key of 256-bit.

The robustness of the proposed ECBFSC is further reinforced by a feedback mechanism, which leads the cipher to a cyclic behavior so that the encryption of each plain pixel depends on the key, the value of the previous cipher pixel and the output of the logistic map.

We have carried out key space analysis, statistical analysis, and key sensitivity analysis to demonstrate the security of the new image encryption procedure. According to the results of our security analysis, we conclude that the proposed ECBFSC is expected to be useful for real-time image encryption and transmission applications.

Furthermore, we have also discussed the characteristics of image cryptosystems, chaos and cryptography including characteristics and analysis of chaotic logistic map, and research issues related to image cryptosystems.

12 References

[1] W. Stallings., "Cryptography and Network Security: Principles and Practice," Prentice-Hall, New Jersey, 1999.

[2] Bruce Schneier, "Applied Cryptography – Protocols, algorithms, and source code in C," John Wiley & Sons, Inc., New York, second edition, 1996.

[3] N. Bourbakis and C. Alexopoulos , Picture data encryption using SCAN patterns. Pattern Recognition 25 6 (1992), pp. 567–581.
[4] Alexopoulos, C., 1989. SCAN, A language for 2-D sequential data accessing. Ph.D. Thesis, University of Patras, Greece.
[5] C.J. Kuo , Novel image encryption technique and its application in progressive transmission. J. Electron. Imaging 24 (1993), pp. 345–351.
[6] Chang, H.K., Liou, J.L., 1994. An image encryption scheme based on quadtree compression scheme. In: Proceedings of the International Computer Symposium, Taiwan, pp. 230–237.
[7] Chin-Chen Chang, Min-Shian Hwang, Tung-Shou Chen, “A new encryption algorithm for image cryptosystems”, The Journal of Systems and Software 58 (2001), 83-91
[8] Fridrich Jiri, Symmetric ciphers based on two dimensional chaotic maps, Int. J. Bifurcat Chaos 8 (1998) (6), pp. 1259–1284.
[9] J. Scharinger, Fast encryption of image data using chaotic Kolmogrov flow, J. Electronic Eng 7 (1998) (2), pp. 318–325.
[10] J.C. Yen, J.I. Guo, A new image encryption algorithm and its VLSI architecture, in: Proceedings of the IEEE workshop signal processing systems, 1999, pp. 430–437.
[11] J.C. Yen, 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, 2000, pp. 49–52.
[12] S. Li, X. Zheng, X. Mou, Y. Cai, Chaotic encryption scheme for real time digital video, Proceedings of the SPIE on electronic imaging, San Jose, CA, USA, 2002.
[13] G. Chen, Y. Mao and C.K. Chui, A symmetric image encryption based on 3D chaotic maps, Chaos Solitons Fractals 21 (2004), pp. 749–761.
[14] W. Diffie and M.E. Hellman , New directions in cryptography. IEEE Trans. Inf. Theory 22 (1976), pp. 644–654.
[15] M. S. Baptista, "Cryptography with chaos". Phys. Lett. A, vol.240, pp.50-54,1998.
[16] G. Jakimoski and L. Kocarev, "Chaos and Cryptography: Block Encryption Ciphers Based on Chaotic Maps," IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications, vol. 48, no. 2, February 2001.
[17] R. Devaney, "An Introduction to Chaotic Dynamical Systems," 2nd ed. Redwood City, CA: Addison-Wesley, 1989.
[18] Steven Henry Strogatz, "Nonlinear dynamics and chaos: With applications to physics, biologym chemistry, and engineering," first ed., Addison-Wesley Publishing Company, Reading, Massachusetts, 1994.
[19] Shujun Li, Guanrong Chen and Xuan Zheng, "Chaos-based encryption for digital images and videos," chapter 4 in Multimedia Security Handbook, February 2004.
[20] Yaobin Mao and Guanrong Chen, "Chaos-based image encryption," in Eduardo Bayro-Corrochano, editor, Handbook of Computational Geometry for Pattern Recognition, Computer Vision, Neural Computing and Robotics. Springer-Verlag, Heidelberg, April 2004.
[21] Yaobin Mao, Guanrong Chen, and Charles K. Chui, "A novel fast image encryption scheme based on 3D chaotic Baker maps," Int. J. Bifurcation and Chaos in June 2003.
[22] Yaobin Mao, Guanrong Chen, and Shiguo Lian, "A symmetric image encryption scheme based on 3D chaotic Cat maps," Chaos, Solitons and Fractals 21, pages 749-761, 2004.