A Novel Image Stream Cipher Based On Dynamic Substitution

Abdelfattah Elsharkawi
Software Engineering,
Communication Engineering
Al-Azhar University
Cairo, Egypt
Sharkawi_eg@yahoo.com

Ragab M. El-Sagheer
Software Engineering,
Communication Engineering
Al-Azhar University
Cairo, Egypt
Relbakar@gmail.com

Haitham Akah
Space Communication
National Authority For
Remote Sensing And Space
Science, Cairo, Egypt
haitham_akah@narss.sci.eg

Hatem Taha
Space Communication
National Authority For
Remote Sensing And Space
Science, Cairo, Egypt
hatem.taha@narss.sci.eg

***Abstract*—Recently, many chaos-based stream cipher algorithms have been developed. Traditional chaos stream cipher is based on XORing a generated secure random number sequence based on chaotic maps (e.g. logistic map, Bernoulli Map, Tent Map etc.) with the original image to get the encrypted image, This type of stream cipher seems to be vulnerable to chosen plaintext attacks. This paper introduces a new stream cipher algorithm based on dynamic substitution box. The new algorithm uses one substitution box (S-box) and a chaotic shuffling process. Each byte in the plain image vector is substituted using a different S-box to get the cipher image vector. This algorithm is designed to be invulnerable to chosen plaintext attacks. In addition, this algorithm is more secured compared to conventional stream cipher.**

Keywords-chaos; encryption; plain image; stream cipher; substitution box.

I. INTRODUCTION

A huge amount of digital data (voices, images and video) is transferred from point to point on various global networks (internet, mobile networks, remote sensing satellites and others). It is necessary to protect such data from unauthorized interception or tampering on the open network. Cryptography is the science that aims to hide (encrypt) the meaning of a message and splits into three main branches: symmetric cipher, asymmetric cipher and protocols [1]. Cryptographic protocols are (roughly speaking), crypto protocols that deal with the application of cryptographic algorithms. Symmetric and asymmetric algorithms can be viewed as building blocks with which applications such as secure internet communication can be realized. The Transport Layer Security (TLS) scheme, which is used in every Web browser, is an example of a cryptographic protocol [2]. Also, it can be defined as a series of steps, involving two or more parties, designed to accomplish a task [2].

Asymmetric algorithms (also called Public-key algorithms) are designed so that the key used for encryption is different from the key used for decryption. These algorithms are called "public-key" because the encryption key can be made public: A complete stranger can use the encryption key to encrypt a message, but only a specific person with the corresponding decryption key can decrypt the message. In these systems, the encryption key is often called the public key, and the decryption key is often called the private key. The private key is sometimes also called the secret key [2]. Symmetric algorithms (sometimes called conventional algorithms) are used where the encryption key can be calculated from the decryption key and vice versa. In most symmetric algorithms, the encryption key and the decryption key are the same. These algorithms are also called secret-key algorithms, single-key algorithms, or one-key algorithms [2]. Symmetric ciphers itself split into stream and block cipher. Block ciphers encrypt an entire block of plaintext bits at a time with the same key. This means that the encryption of any plaintext bit in a given block depends on every other plaintext bit in the same block. In practice, the vast majority of block ciphers either has a block length of 128 bits (16 bytes) such as the advanced encryption standard (AES), or a block length of 64 bits (8 bytes) such as the data encryption standard (DES) or triple DES (3DES) algorithm. Stream ciphers encrypt bits or bytes individually. This is achieved by adding a bit or byte from a key stream to a plaintext stream. There are synchronous stream ciphers where the key stream depends only on the key and asynchronous ones where the key stream also depends on the cipher text. If the dotted line in Figure 1 is present, the stream cipher is an asynchronous one. Most practical stream ciphers are synchronous ones.

When dealing with cryptographic applications, linear methods for generating pseudo-random sequences like linear feedback shift registers (LFSRs), linear conjugate gradient (LCGs) or their proper combinations are highly not recommended, since efficient algorithms are at disposal to predict the sequence on the basis of a relatively short sequence observation [3]. Chaos is a fascinating phenomenon that has been observed in nature (weather and climate, dynamics of satellites in the solar system, time evolution of the magnetic field of celestial bodies, and population growth in ecology) and laboratory (electrical circuits, lasers, chemical reactions, fluid dynamics, mechanical systems, and magneto-mechanical devices and others). Due to the inherent properties of chaotic systems such as statistical ergodicity, pseudo-randomness, and sensitivity to initial conditions and control parameters, the chaotic systems has been extensively applied in cryptography

[4]. S-boxes have been widely used as a base of new encryption strategies due to their properties such as nonlinearity, differential uniformity, and strict avalanche criterion. Recently, the use of S-box became popular in image ciphers as a main approach to performing substitution. In this paper a new proposed stream cipher for the images is introduced based on chaotic substitution box which can be considered as a basic nonlinear component of symmetric key algorithms.

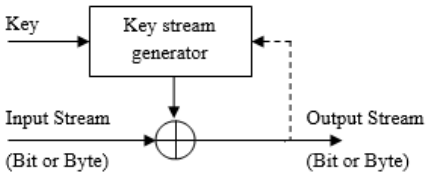

Fig. 1. Synchronous and asynchronous stream ciphers

II. RELATED WORK

Recently many chaotic key based image stream ciphers have been proposed. In [5] a chaotic key-based design for image encryption and decryption in which the gray level of each pixel is XORed or XNORed bit-by-bit to one of the two predetermined keys was introduced. This paper kept focus on three main factors i.e. high security, low computation and no distortion. In [6], an image algorithm that encodes digital images pixel by pixel was proposed. This algorithm then uses stream cipher that encrypts each pixel using a special mathematical set of functions known as the key. This algorithm uses the same key at both the sending and the receiving ends.

Some image encryption algorithms have been also recently proposed based on chaotic substitution. In [7], an image encryption algorithm based on the theory of S-box in advanced encryption standard (AES) was proposed. Thirty different S-boxes structured by different irreducible polynomials were used. Each element in the S-boxes is numbered from 0 to 255, and is one to one mapped. The algorithm uses logistic map which is based on a chaotic map as one S-box of the 30 S-boxes will be used in the byte substitution process until finishing the whole encrypting image. In [8], an image encryption algorithm in which an external 256-bit key is used while the last pixel of plain image is used to generate the parameters and the initial states of the chaotic systems for the first S-box was proposed. The plain image is divided into groups in which the pixels are substituted by S-boxes. The image pixels are divided to several groups according to rows and adjacent rows are collected in different groups. For each group, a new S-box is generated and used. By this way, the corrections between vertical adjacent pixels are smashed. And then the same method on columns in order to smash its corrections between horizontal adjacent pixels is used. In [9], an image encryption algorithm based on circular substitution box and key stream buffer was suggested. The S-box is considered as a circular sequence with a head pointer in this case, and each image pixel is replaced with an element of an S-box according to both the pixel value and the head pointer, the head pointer varies with the previous substituted pixel. It is found that some S-box-only ciphers are vulnerable to chosen plaintext attacks. In [10], the security issues for S-box-only image ciphers and presented a successful cryptanalysis was carefully studied.

III. PROPOSED ALGORITHM

Image data have special properties such as bulk capacity, high redundancy and high correlation among the pixels that imposes special requirements on any encryption technique [11]. The proposed algorithm was designed to match the image special properties and get a high speed processing on the hardware devices such as FPGA or ASIC. The inter-pixel displacement or shifting of the image pixels, where pixels are completely moved from its position to a new position has been used in image encryption [12, 13]. In the proposed algorithm such displacement shall be applied to the S-box, where the positions of a dynamic substitution box contents will be used in displacement rather than using inter pixel displacement of images.

A dynamic (16*16) S-box is used to substitute the input byte (image pixel) by another one, based on the input image byte and the logistic map output. This map is used to generate the chaotic keys which are used to shuffling the substitution box contents to get a dynamic substitution box.

$$X_n = r * X_{n-1} * (1 - X_{n-1}) \quad (1)$$

Where r is the control parameter. The logistic map represents nothing more than an idealized population model. The proposed algorithm consists of four main blocks as shown in Figure 2. The four blocks are the Input (Initial) Substitution box, the Shuffling Keys generator block (e.g. two chaotic generators of logistic maps may be used to generate two keys, namely Row Key (*RKey*) and Column Key (*CKey*), the Shuffling block and the Substitution stage (output S-box).

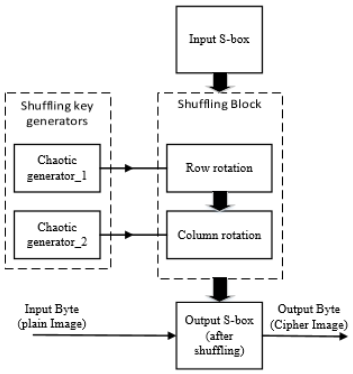

Fig. 2. Block diagram of proposed algorithm

A. Encryption steps

- The plain image (indicated early or real stream) will be constructed as a stream vector of bytes, assuming a plain image P has N bytes, and then having

$$P = \{P_0, P_1, \ldots, P_{N-1}\} \quad (2)$$

- For each plain image byte, there are two 64 bit keys (*RKey*, *CKey*) which will be generated from logistic map generators, forming:

$$RKey = \{RKey_0, RKey_1, \ldots, RKey_{N-1}\} \quad (3)$$

$$CKey = \{CKey_0, CKey_1, \ldots, CKey_{N-1}\} \quad (4)$$

- Each generated key is divided into 4 bit sub rotation keys to get (16 row keys) + (16 column keys).
- Initial S-box matrix (16 * 16) is constructed as shown in Figure 3. Each element in the S-box is numbered from 0 to 255, and is one to one mapped.

$S_{1,1}$	$S_{1,2}$	$S_{1,3}$	$S_{1,4}$	$S_{1,5}$	$S_{1,6}$	$S_{1,7}$	$S_{1,8}$	$S_{1,9}$	$S_{1,10}$	$S_{1,11}$	$S_{1,12}$	$S_{1,13}$	$S_{1,14}$	$S_{1,15}$	$S_{1,16}$
$S_{2,1}$	$S_{2,2}$	$S_{2,3}$	$S_{2,4}$	$S_{2,5}$	$S_{2,6}$	$S_{2,7}$	$S_{2,8}$	$S_{2,9}$	$S_{2,10}$	$S_{2,11}$	$S_{2,12}$	$S_{2,13}$	$S_{2,14}$	$S_{2,15}$	$S_{2,16}$
$S_{3,1}$	$S_{3,2}$	$S_{3,3}$	$S_{3,4}$	$S_{3,5}$	$S_{3,6}$	$S_{3,7}$	$S_{3,8}$	$S_{3,9}$	$S_{3,10}$	$S_{3,11}$	$S_{3,12}$	$S_{3,13}$	$S_{3,14}$	$S_{3,15}$	$S_{3,16}$
$S_{4,1}$	$S_{4,2}$	$S_{4,3}$	$S_{4,4}$	$S_{4,5}$	$S_{4,6}$	$S_{4,7}$	$S_{4,8}$	$S_{4,9}$	$S_{4,10}$	$S_{4,11}$	$S_{4,12}$	$S_{4,13}$	$S_{4,14}$	$S_{4,15}$	$S_{4,16}$
$S_{5,1}$	$S_{5,2}$	$S_{5,3}$	$S_{5,4}$	$S_{5,5}$	$S_{5,6}$	$S_{5,7}$	$S_{5,8}$	$S_{5,9}$	$S_{5,10}$	$S_{5,11}$	$S_{5,12}$	$S_{5,13}$	$S_{5,14}$	$S_{5,15}$	$S_{5,16}$
$S_{6,1}$	$S_{6,2}$	$S_{6,3}$	$S_{6,4}$	$S_{6,5}$	$S_{6,6}$	$S_{6,7}$	$S_{6,8}$	$S_{6,9}$	$S_{6,10}$	$S_{6,11}$	$S_{6,12}$	$S_{6,13}$	$S_{6,14}$	$S_{6,15}$	$S_{6,16}$
$S_{7,1}$	$S_{7,2}$	$S_{7,3}$	$S_{7,4}$	$S_{7,5}$	$S_{7,6}$	$S_{7,7}$	$S_{7,8}$	$S_{7,9}$	$S_{7,10}$	$S_{7,11}$	$S_{7,12}$	$S_{7,13}$	$S_{7,14}$	$S_{7,15}$	$S_{7,16}$
$S_{8,1}$	$S_{8,2}$	$S_{8,3}$	$S_{8,4}$	$S_{8,5}$	$S_{8,6}$	$S_{8,7}$	$S_{8,8}$	$S_{8,9}$	$S_{8,10}$	$S_{8,11}$	$S_{8,12}$	$S_{8,13}$	$S_{8,14}$	$S_{8,15}$	$S_{8,16}$
$S_{9,1}$	$S_{9,2}$	$S_{9,3}$	$S_{9,4}$	$S_{9,5}$	$S_{9,6}$	$S_{9,7}$	$S_{9,8}$	$S_{9,9}$	$S_{9,10}$	$S_{9,11}$	$S_{9,12}$	$S_{9,13}$	$S_{9,14}$	$S_{9,15}$	$S_{9,16}$
$S_{10,1}$	$S_{10,2}$	$S_{10,3}$	$S_{10,4}$	$S_{10,5}$	$S_{10,6}$	$S_{10,7}$	$S_{10,8}$	$S_{10,9}$	$S_{10,10}$	$S_{10,11}$	$S_{10,12}$	$S_{10,13}$	$S_{10,14}$	$S_{10,15}$	$S_{10,16}$
$S_{11,1}$	$S_{11,2}$	$S_{11,3}$	$S_{11,4}$	$S_{11,5}$	$S_{11,6}$	$S_{11,7}$	$S_{11,8}$	$S_{11,9}$	$S_{11,10}$	$S_{11,11}$	$S_{11,12}$	$S_{11,13}$	$S_{11,14}$	$S_{11,15}$	$S_{11,16}$
$S_{12,1}$	$S_{12,2}$	$S_{12,3}$	$S_{12,4}$	$S_{12,5}$	$S_{12,6}$	$S_{12,7}$	$S_{12,8}$	$S_{12,9}$	$S_{12,10}$	$S_{12,11}$	$S_{12,12}$	$S_{12,13}$	$S_{12,14}$	$S_{12,15}$	$S_{12,16}$
$S_{13,1}$	$S_{13,2}$	$S_{13,3}$	$S_{13,4}$	$S_{13,5}$	$S_{13,6}$	$S_{13,7}$	$S_{13,8}$	$S_{13,9}$	$S_{13,10}$	$S_{13,11}$	$S_{13,12}$	$S_{13,13}$	$S_{13,14}$	$S_{13,15}$	$S_{13,16}$
$S_{14,1}$	$S_{14,2}$	$S_{14,3}$	$S_{14,4}$	$S_{14,5}$	$S_{14,6}$	$S_{14,7}$	$S_{14,8}$	$S_{14,9}$	$S_{14,10}$	$S_{14,11}$	$S_{14,12}$	$S_{14,13}$	$S_{14,14}$	$S_{14,15}$	$S_{14,16}$
$S_{15,1}$	$S_{15,2}$	$S_{15,3}$	$S_{15,4}$	$S_{15,5}$	$S_{15,6}$	$S_{15,7}$	$S_{15,8}$	$S_{15,9}$	$S_{15,10}$	$S_{15,11}$	$S_{15,12}$	$S_{15,13}$	$S_{15,14}$	$S_{15,15}$	$S_{15,16}$
$S_{16,1}$	$S_{16,2}$	$S_{16,3}$	$S_{16,4}$	$S_{16,5}$	$S_{16,6}$	$S_{16,7}$	$S_{16,8}$	$S_{16,9}$	$S_{16,10}$	$S_{16,11}$	$S_{16,12}$	$S_{16,13}$	$S_{16,14}$	$S_{16,15}$	$S_{16,16}$

Fig. 3. Initial S-box

- Perform row shuffling (rotation) using *RKey* for the S-box (as indicated).
- After finishing the row rotation, an intermediate S-box will be released.
- Perform column shuffling (rotation) using *CKey* for the intermediate S-box (after row shuffling).
- After shuffling the initial S-box using row and column rotations, the newly constructed S-box will be used to get the encrypted value and will also be used as input S-box for the next generation samples (rotation keys).
- The previous steps will be repeated until the end of the plain image.

B. Decryption steps

The decryption steps are similar to the encryption steps. They use cipher image instead of plain image and inverse S-box instead of S-box but the same chaotic generators are used with the inverse of the initial S-box. Due to random shuffling block, there are exhaustive number of S-boxes that can be constructed and used to substitute the plain image. Additionally, the encrypted image is sensitive to the initial S-box which make the proposed algorithm robust to the chosen plaintext attacks and suitable for (huge images) such as remote sensing satellite images and video streams. Hardware implementation can be used to increase the encryption speed, and hence realizing a real time algorithm.

IV. Results and Discusion

A good encryption algorithm should be robust against most kinds of known cryptanalytic, statistical and brute-force attacks. This section discusses the security analysis of the proposed algorithm such as statistical analysis, sensitivity analysis with respect to the key and key space analysis. To prove that the proposed cryptosystem is secure against most common attacks [4, 5], a Simulation was carried out using national instruments software (NI Lab VIEW 2013) to encrypt a number of images using the proposed algorithm, and some calculations were performed as following.

A. Statistical Analysis

1) Entropy

Entropy is a cumulative measure of the frequency of the intensity levels in an image. Due to the characteristics of the human eye, which is insensitive to high frequency components, an image of high entropy is not visually perceivable. Entropy is given by

$$h = -\sum_i \left(p_i \log_2 \left(\frac{1}{p_i} \right) \right) \quad (5)$$

Where P_i is the frequency of intensity level i in the image, the maximum h for 8–bit image can attain is 8, and the average of our results is ($h > 7.999$). Hence a statistical attack is very difficult to make. The entropies for different images are listed in Table I.

TABLE I. ENTROPY

Image Sample	Original Entropy	Entropy of [8]	Entropy of [14]	Proposed algorithm
Lena	7.44557	7.9971	7.9870	7.99937
baboon	7.35778	N.A.	N.A.	7.99931
Fixed value(125)	0.0	N.A.	N.A.	7.99928

2) Correlation Coefficient Analysis

The correlation analysis is the most fundamental method used in determining the similarity between two images, especially in encryption applications. In order to evaluate the overall similarity between plain image and the encrypted image, the cross correlation between the two images will calculated. This is typically done at every step by subtracting the mean and dividing by the standard deviation. Thus, the cross-correlation of an image $F(x, y)$ with an image $T(x,y)$ is

$$\frac{1}{n} \sum_{x,y} \frac{\left(F(x,y)-\overline{F}\right)\left(T(x,y)-\overline{T}\right)}{\sigma F \sigma T} \quad (6)$$

Where n is the number of pixels in F and T, $\overline{F}$ and $\overline{T}$ are the averages of $F(x, y)$ and $T(x, y)$ respectively, σF is the standard deviation $F(x ,y)$, and σT is the standard deviation of $T(x, y)$ The correlation coefficient between the original image and the encrypted image is shown in Table II.

TABLE II. CORRELATION COEFFICIENT

Image Sample	[14]	Proposed algorithm
Lena	-0.0086	-0.00064
baboon	N.A.	-0.00271
Peppers	N.A.	0.00040

3) Histogram

Referring to the plot of gray levels in an image against their frequencies of occurrence, the following figures show that the histogram is uniform and all gray levels have the same frequency of occurrence with the same probability. The histogram of the cipher image has no statistical relation to the plain image and hence does not provide any clue for a statistical attack on the proposed encryption scheme as shown in Figures 4-6.

B. Sensitivity Analysis

1) Key Space Analysis

In this algorithm, the key space depends on the chaotic map or maps which will be used in shuffling key generators. For example two logistic maps have been used for shuffling the substitution box. One for rows and the other for columns. Logistic map equations are

$$X_n = r_1 * X_{n-1} * (1 - X_{n-1}) \quad (7)$$

$$Y_n = r_2 * Y_{n-1} * (1 - Y_{n-1}) \quad (8)$$

Where r_1, r_2 are represented in 60 bits and the initial values X_n, Y_n are represented in 64 bits value, while the least 4 bits were neglected to increase the key sensitivity. Then the actual values are X_n = 60 bits and Y_n = 60 bits. The number of key bits are = (Kx + Ky) = 60 + 60 + 60 + 60 = 240 bits which is an acceptable key space in symmetric key encryption in modern cryptography.

2) Key sensitivity

A perfect image encryption algorithm should be sensitive to small changes in the secret key. So 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 algorithm, the following steps were performed:

- Two keys with only one bit difference have been chosen to encrypt the same plain baboon image (512 * 512) pixel. The two keys written in hexadecimal form the following code: *K1*=(*Kx*+*Ky*)=(123456789ABCDEF123456789ABCDEF+123456789ABCDE**F**F123456789ABCDE)

  and

  K2=(*Kx*+*Ky*)=(123456789ABCDEF123456789ABCDEF+123456789ABCDE**E**F123456789ABCDE) so, there are two encrypted images *i1* with *K1* and *i2* with *K2*.
- The encrypted images *i1* and *i2* were compared by calculation of the cross correlation between them.

The above steps were repeated using different plain images with the same keys as shown in Table III.

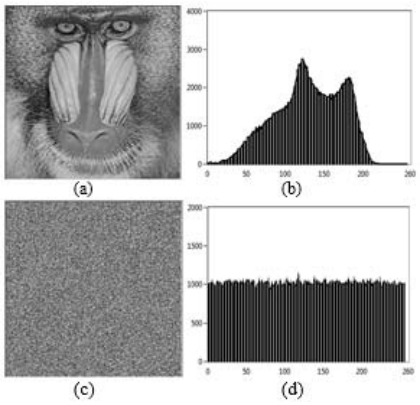

Fig. 4. Histograms: (a) baboon image, (b) histogram of plain baboon, (c) baboon encrypted image, (d) histogram of encrypted baboon.

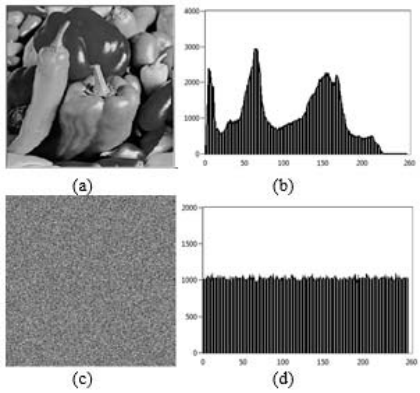

Fig. 5. Histograms: (a) peppers image, (b) histogram of plain peppers, (c) peppers encrypted image, (d) histogram of encrypted peppers.

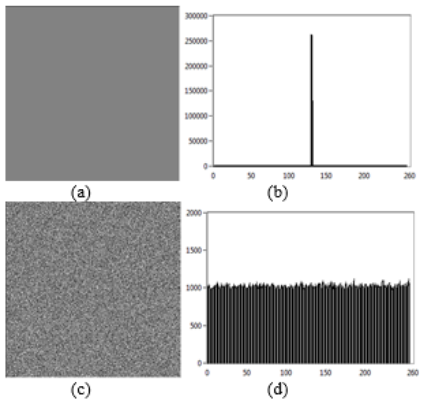

Fig. 6. Histograms: (a) fixed value image, (b) histogram of plain fixed value, (c) fixed value encrypted image, (d) histogram of encrypted fixed value.

TABLE III. KEY SENSITIVITY

Image Sample	Cross Correlation between encrypted images using (*K1*) and (*K2*)
Baboon	-4.88697E-5
Lena	0.000932412
Fixed value(125)	0.000476637

V. CONCLUSION

This paper proposed an image encryption algorithm based on a dynamic chaotic substitution box based on logistic maps. Statistical results using entropy, correlations, and histogram show very good results for the proposed algorithm. Simulation results show that it is also robust against most attacks due to the strong security, the key space, the initial S-box and the exhaustive number of S-boxes used in substitutions. The algorithm is so simple to be easily and efficiently implemented using hardware implementations such as FPGA or ASIC chips. There is also no size limit to the input image because the algorithm works as an ergodic stream cipher.

REFERENCES

[1] C. Paar, Ing. J. Pelzl, Understanding Cryptography, Springer, 2010

[2] B. Schneier, Applied Cryptography, Second Edition, John Wiley & Sons, Inc., 1996

[3] J. Boyar, "Inferring sequences produced by pseudo-random number generators", J. ACM, Vol. 36, No. 1, pp. 129–141, 1989

[4] J. Won, H. Kim, "Commun Nonlinear Sci Numer Simulat An image encryption scheme with a pseudorandom permutation based on chaotic maps", Commun. NONLINEAR Sci. Numer. Simul., Vol. 15, No. 12, pp. 3998-4006, 2010

[5] J. Yen, J.Guo, "A New Chaotic Key-Based Design for Image Encryption and Decryption", IEEE International Symposium on ISCAS 2000, Geneva, pp. IV-49-IV-52, 2000

[6] A. Kaushik, S. Khanna, M. Barnela, A. Kumar, "Stream Encryption Standard for Digital Images", International Journal of Computer and Electrical Engineering, Vol. 3, No. 2, pp. 240-243, 2011

[7] W. De, Z. Yuan-Biao, "Image encryption algorithm based on S-boxes substitution and chaos random sequence", 2009 Int. Conf. Comput. Model. Simulation, ICCMS 2009, pp. 110–113, 2009

[8] X. Wang, Q. Wang, "A novel image encryption algorithm based on dynamic S-boxes constructed by chaos", Nonlinear Dyn., Vol. 75, No. 3, pp. 567–576, 2013

[9] X. Zhang, Z. Zhao, J. Wang, "Chaotic image encryption based on circular substitution box and key stream buffer", Signal Process. Image Commun., Vol. 29, No. 8, pp. 902–913, 2014

[10] Y. Zhang, D. Xiao, "Cryptanalysis of S-box-only chaotic image ciphers against chosen plaintext attack", Nonlinear Dyn., Vol. 72, No. 4, pp. 751–756, 2013

[11] Eduardo Bayro Corrochano, Handbook of Geometric Computing, Springer, 2005

[12] R. Siwatch, V. Kumar, "Image Encryption Based on Inter-Pixel Displacement", International Journal of Scientific Research Engineering & Technology, Vol. 3, No. 3, pp. 673–676, 2014

[13] R. Mathews, M. Iaeng, A. Goel, M. Iaeng, P. Saxena, V. P. Mishra, "Image Encryption Based on Explosive Inter-pixel Displacement of the RGB Attributes of a PIXEL", Proceedings of the World Congress on Engineering and Computer Science, Vol. I, pp. 19–22, 2011

[14] B. Aïssa, D. Nadib, R. Mohamedc, "Image Encryption Using Stream Cipher Based on Nonlinear Combination Generator with Enhanced Security", New Trends Math. Sci., Vol. 1, No. 1, pp. 10–19, 2013