Chaotic Block Image Scheme using Large Key Space and Message Digest Algorithm

Fahmi Nasser Ali *fahminasser_88@yahoo.com*
Department of information technology
University of Aden
Aden, Yemen

Mussa Mohamed Ahmed *mussa_m7@yahoo.com*
Department of electronics and communication
University of Aden
Aden, Yemen

Abstract

In this paper, chaotic block image scheme using large key space and message digest algorithm**.** Cat map intended for confusion and 2D-Sine Tent Composite map (2D-STCM) key generator intended for diffusion. Confusion is implemented by 2D Cat map with arbitrary block size. In the first tendency, 2D cat map use for local shuffling of indexes inside blocks, while in the second tendency, 2D cat map used for global shuffling of whole image indexes. The designed algorithm executes two confusions and one diffusion in each iteration. To increase the security level, the message digestion algorithm is used as a fingerprint for the plain image that creates the initial value of the key. After that 2D-STCM generates a large key stream. Diffusion implementation takes place by XOR operation; between a key stream and confused image. Experimental results, show that security level increases due to integration of confusion and diffusion. On the other side large key space and the high sensitivity of secret keys have been given a guarantee for the performance of the security. Performance measures reach to the top value among those in the similar researches. To verify the obtained results, authors implemented inverse chaos. All the tests are processed by MATLAB 2015a.

Keywords: Image Encryption, 2D-STCM, Cat Map, MD5, Confusion, Diffusion.

1. INTRODUCTION

The converting of information image, video, audio and text from insecure form into secure form and transmitting it over the insecure network is known as cryptography [2]. In cryptography, there are many traditional encryption schemes such as Advanced Encryption Standard (AES), Data Encryption Standard (DES), 3DES and Rivest-Shamir-Adleman algorithm (RSA) etc. [2-3]. Traditional encryption schemes are not suitable for image encryption because there are inherent features in image like bulk data capacity, correlation among adjacent pixels and high redundancy so the image encryption differs from text encryption.

System Chaos is the best in encryption and decryption for multimedia. Chaos system was discovered by Edward Lorenz in 1963 [5]. Chaos system properties are pseudo-random, non-periodicity, ergodicity and highly sensitive to system parameters and initial conditions [6]. Chaos systems are based on initial condition and control parameter. Therefore, a little change in initial condition or control parameter cause a completely change. Chaotic maps can be categorized into two groups: 1D chaotic maps such as Logistic map, Sine map and Tent map etc. and a high-dimensional chaotic map such as Arnold map, Exponential map and Henon map etc. The combining of chaotic theory and cryptography are important fields in multimedia security to achieve high security. Therefore, the level of security is high when it achieves grouping of confusion and diffusion [7]. The concept of confusion is to change the pixel's position of plain image and diffusion is to change the gray level value of confused image.

Rest of the paper is discussed as follows. Section 2 explores the related work in the area of chaotic block Image scheme using large key space and message digest algorithm, discuss

Arnold Cat Map, MD5 and 2D-STCM. Section 3 represents a detail description of proposed designed scheme. Section4 introduces the experimental results, compares varies image encryption techniques and discusses the outcomes from several algorithms. Section5 is conclusion and future work.

2. RELATED WORK

There are many researches in digital image encryption based on chaotic maps and key stream generators. In [8], cat map and significant bit-planes are executed. The cat map is used to scramble the positions of image pixels and the significant bit-planes can diffuse the image pixels, but is not robust against differential attack because NPCR is not approaching to 100%and UACI is not approaching to 33% In [9], Arnold cat map and 1D Logistic map are applied. The Arnold cat map is used to shuffle the pixels' positions of image and the 1D Logistic map diffused the image pixels to get encrypted image, but is not robust against brute-force attack because key space of encryption algorithm is not large. In [10], the three-dimensional Henon chaotic map and the Cat chaotic map are executed. The cat chaotic map is used to confuse the positions of image pixels and the three-dimensional Henon chaotic map can diffuse the image pixels. Then, repeat the confusing and diffusing process to increase the resistance to statistical and related attacks. The disadvantages of this method are the low encryption speed. In [11], the 2D-sine tent composition map (2D-STCM) and the chaotic circular pixel shuffling (CCPS) are applied. 2D-STCM is used to generate a key sequence and CCPS is used to shuffle the pixels' positions of image based on 2D-STCM. In [12], algorithm based on one-time keys and robust chaotic maps. Generate pseudo-random key sequence by a piecewise linear chaotic map. The initial conditions were generated by the MD5 of the mouse positions. This method is robust against brute-force attack because key space of encryption algorithm is very large, but is not strong in resisting statistical attack because correlation coefficients of ciphered image is not good. In [16], created new chaotic maps based on Beta function. It executes three phases: permutation, diffusion and substitution. In [17], new chaotic map, strong S-boxes and permutation function are implemented. It executes four phases: diffusion, substitution, diffusion and permutation. The diffusion process is based on a new chaotic map. The substitution process is based on a strong S-boxes. The permutation process is based on a permutation function. In [19], cat map, logistic and tent maps are applied. The cat map is used to permutation process, and then the logistic and tent maps diffused the image pixels to get the encrypted image, but is not robust against statistical attack because correlation coefficients of ciphered image is not good. In [20], combined 2-D Sine Logistic modulation map, derived from the Logistic and Sine maps with a chaotic magic transform, aim was to resist various attacks but, without mention target or specific types. In [22], logistic map, cat map, SHA-3 hash function and auto-updating system are executed. Algorithm acts like one-time pad. The logistic map and SHA-3 hash function are used to permutation process, and then the cat map and SHA-3 hash function diffused the image pixels to get the encrypted image, but it is not strong against statistical attack because correlation coefficients of ciphered image are not too low.

In this paper, the idea of encryption scheme proposes to achieve confusion and diffusion. First, the whole image is divided into blocks, random change of block size in each iteration. Then, Arnold cat map is Implemented to obtain confused image. Second, the key stream is generated by 2D-STCM. Then, implement bit-XOR operation between key stream of 2D-STCM and confused image. the previous steps are repeated two or more times to obtain the final encrypted image.

Below Arnold Cat Map, MD5 and 2D-STCM Key are analyzed.

2.1 Arnold Cat Map

Arnold Cat Map [9] is two-dimensional discrete map. This map is used to shuffle the pixel's positions of image.

$$\begin{bmatrix} X' \\ Y' \end{bmatrix} = \begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \bmod(N) \qquad (1)$$

The Arnold cat map uses two parameters (a, b); the a, b are control parameters which change the pixels' positions in image, where (a, b) are positive integers stand for the control parameters, (x, y) are the old pixel position and (x', y') are the new pixel position.

The disadvantage of Arnold map, that the same original image appears after several iterations; which concludes that Arnold map has periodic behavior. To solve this, issue the parameters (a, b) are generated randomly by using 2D-STCM or manual change of parameters (a, b) in each iteration.

2.2 Message Digest Algorithm

The message digest algorithm (MD5) takes any arbitrary length as input data, and produces output of 128-bit (32-digit hexadecimal) as fingerprint [15]. Each message produces a special MD5. It has been widely used to verify data or image integrity. As can be seen, in proposed scheme Fig.2, MD5 produces initial value. More details, explanations and applications of MD5 are given in section 3.1.

2.3 Two Dimensional-Sine Tent Composite Map

The 2D-Sine Tent composite map [11] is a discrete-time dynamical system. It is built from 1D Sine map and 1D Tent map defined in equations (3) and (4) respectively:

$$x_{n+1} = \mu \times \sin(\pi x_n) \tag{3}$$

$$x_{n+1} = \begin{cases} rx_n & . \ 0 \le x < 0.5 \\ r(1-x_n) & . \ 0.5 \le x \le 1 \end{cases} \tag{4}$$

Here μ and r are control parameters lies in the range (0, 4] [11]. To overcome the limitations of 1D chaotic map, combined Sine and Tent maps. It can be defined as:

$$x_{n+1} = \begin{cases} \left((\sin(\pi y_n) + 3) \times \frac{x_n}{2}\right) mod\ 1. & x_i < 0.5 \\ \left((\sin(\pi y_n) + 3) \times \frac{(1-x_n)}{2}\right) mod\ 1 & x_i \ge 0.5 \end{cases} \tag{5}$$

$$y_{n+1} = \begin{cases} \left((\sin(\pi x_n) + 3) \times \frac{y_n}{2}\right) mod\ 1. & y_i < 0.5 \\ \left((\sin(\pi x_n) + 3) \times \frac{(1-y_n)}{2}\right) mod\ 1. & y_i \ge 0.5 \end{cases} \tag{6}$$

A trajectory of 2D-Sine Tent composite map is excellent chaotic behavior. Figure 1 compares the trajectories of 2D-STCM, 2D-SLMM and 2D-LM [11]. From trajectory figure1(a) of 2D-STCM covers a large region compared with other two. So, it proves that the proposed chaotic function has better ergodicity and randomness property.

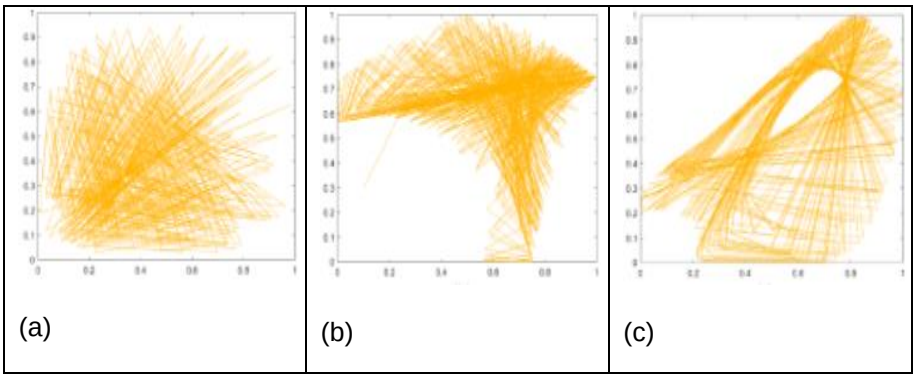

FIGURE 1: Trajectories of (**a**) 2D-STCM Map; (**b**) 2D-SLMM Map and (**c**) 2D-Logistic Map [11].

3. PROPOSED DESIGNED SCHEME

The proposed image encryption scheme has two or more iterations, in each iteration there is group of confusion and diffusion processes. In first confusion process, an Image of size N×N is random decomposed into M×M block, 2D-cat map is used for local shuffling of indexes inside blocks. In second confusion process, 2D-cat map is used for global shuffling of whole image indexes. In diffusion process, XOR operation between key stream {Ki} of 2D-STCM and confused image is implemented.

3.1 Initial Value (X) Generation using MD5

Steps to generate the initial value (x): First, extract the value of MD5 hexadecimal from plain image. Second, convert the hexadecimal MD5 value to a decimal value. Third, use the summation operation to sum decimal values to obtain a single value. Finally, convert the decimal value to a fractional value by dividing by 1000.

Let's take a look at this example, on a system that has an image called " Lena.TIF " the MD5 hash would look as follows:

Image name	Hash Value
C:\ Lena.TIF	5649b212f4f9e5c781a9e4f8c75208e5

Hexadecimal Value	Decimal Value
5649b212f4f9e5c781a9e4f8c75208e5	5.6.4.9.11.2.1.2.15.4.15.9.14.5.12.7.8. 1.10.9.14.4.15.8.12.7.5.2.0.8.14.5

Sum (Decimal Value) = 243. Initial value (x) = 243/10000 = 0.0243.

As shown in Fig.2 **y** as second initial value could be given by user directly or by using indirect different methods from given image.

3.2 Key Sequence Generation using 2D-STCM

The equations (5) and (6) of 2D-STCM are repetitive for generate two key sequences {K1} and {K2} which as original image size. The initial values (x, y) are a fractional value which passes to the 2D-STCM equations to generate two keys sequences. In the next iteration, we change the initial values (x, y). The random change of initial values (Δx, Δy) in each iteration increasing the key space that resist brute-force attack. In [14], shows preprocessing of fractional key to obtain integer key with 8 bits suitable for image encryption as following:

$$x_n = \left(\text{floor}\left(x_n \times 2^{32}\right)\right) \bmod 256. \qquad x_n = k_1 \qquad (7)$$

$$y_n = \left(\text{floor}\left(y_n \times 2^{32}\right)\right) \bmod 256. \qquad y_n = k_2 \qquad (8)$$

where Floor (x_n, y_n) returns the values of (x_n, y_n) to the nearest integer and mod (x_n, y_n) returns the rest of the division. Preprocess leads to a better distribution.

3.3 Image Encryption Process

In this section, the proposed image encryption process is explained.

Step 1: Read two dimensional 8-bit gray scale image $\{I_{i\times j}\}$ of size N×N, where N = (256, 512 or 1024).
Step 2: Decompose the whole image $\{I_{i\times j}\}$ into the M×M size blocks, where M = (64, 16,128…).
Step 3: Apply a 2D-cat map with each block using the control parameters (a_i , b_i) to shuffle the pixel's positions with each block and perform this step two or more times to get first confused image $I_{1_{(i\times j)}}$.

Step 4: Apply the 2D-cat map with first confused image $\{I_{1_{(i\times j)}}\}$ using the control parameters $(a_i\ ,\ b_i)$ to shuffle the pixel's positions of first confused image and perform this step two or more times to get a second confused image $\{I_{2_{(i\times j)}}\}$.

Step 5: In diffusion process. The 2D-STCM keys stream $\{k_1,\ k_2\}$ are converted to two dimensions $\{k_{1_{(i\times j)}},\ k_{2_{(i\times j)}}\}$ as the size of confused image. Then select one key used in the diffusion process. Final, apply a bit-XOR operation between one key stream $\{k_{1\ or\ 2_{(i\times j)}}\}$ from 2D-STCM and the second confused image $\{I_{2_{(i\times j)}}\}$, to get the final encrypted image.

$$\{E_{(i\times j)}\} = \{I_{2_{(i\times j)}}\} \oplus \{k_{1\ or\ 2_{(i\times j)}}\} \quad (9)$$

Step 6: In the next iterations, the previous steps are repeated to obtain the encrypted image.

3.4 Image Decryption Process

In this section decryption process of the encrypted image is explained.

Step 1: In diffusion process. 2D-STCM keys stream $\{k_1,\ k_2\}$ are converted to two dimensions $\{k_{1_{(i\times j)}},\ k_{2_{(i\times j)}}\}$ as the size of encrypted image. Then chose the same key that was used in the encryption process, it should be used in the decryption process. Apply bit-XOR operation between key stream $\{k_{1\ or\ 2_{(i\times j)}}\}$ from 2D-STCM and the encrypted image $\{E_{(i\times j)}\}$, to get the second confused image $\{I_{2_{(i\times j)}}\}$.

In diffusion process. The 2D-STCM keys stream $\{k_1,\ k_2\}$ are converted to two dimensions $\{k_{1_{(i\times j)}},\ k_{2_{(i\times j)}}\}$ as the size of confused image. Then select one key used in the diffusion process. Final, apply a bit-XOR operation between one key stream $\{k_{1\ or\ 2_{(i\times j)}}\}$ from 2D-STCM and the second confused image $\{I_{2_{(i\times j)}}\}$, to get the final encrypted image.

$$\{I_{2_{(i\times j)}}\} = \{E_{(i\times j)}\} \oplus \{k_{1\ or\ 2_{(i\times j)}}\} \quad (10)$$

Step 2: Apply the 2D-cat map with second confused image $\{I_{2_{(i\times j)}}\}$ using the control parameters $(a_i\ ,\ b_i)$ to shuffle the pixel's positions of second confused image and perform this step two or more times to get a first confused image $\{I_{1_{(i\times j)}}\}$.

Step 3: Decompose the first confused image $I_{1_{(i\times j)}}$ into the M×M size blocks, where M = (64, 16,128,).

Step 5: Apply a 2D-cat map with each block using the control parameters $(a_i\ ,\ b_i)$ to shuffle the pixel's positions with each block and perform this step two or more times to get a decrypted image.

Step 6: In the next iterations, the previous steps are repeated to obtain original image.

Looking at the left-hand side of the figure 2, we can see that the processing of the key in three phases. First, put an initial value (x, y) as a fractional value. Second, random change of the initial values (x, y) in each iterate. Finally, the initial values (x, y) enter to 2D-STCM equations which generate a two sequence keys as the size of the encryption image. These keys are used in the cat map and diffusion process. The right-hand portion of the figure 2, shows the encryption steps in four phases. First, divided the whole image into the blocks. Second, Apply a 2D-cat map with each block. Third, apply a 2D-cat map with whole image. Finally, implement a bit-XOR operation between key stream {Ki} of 2D-STCM and confused image.

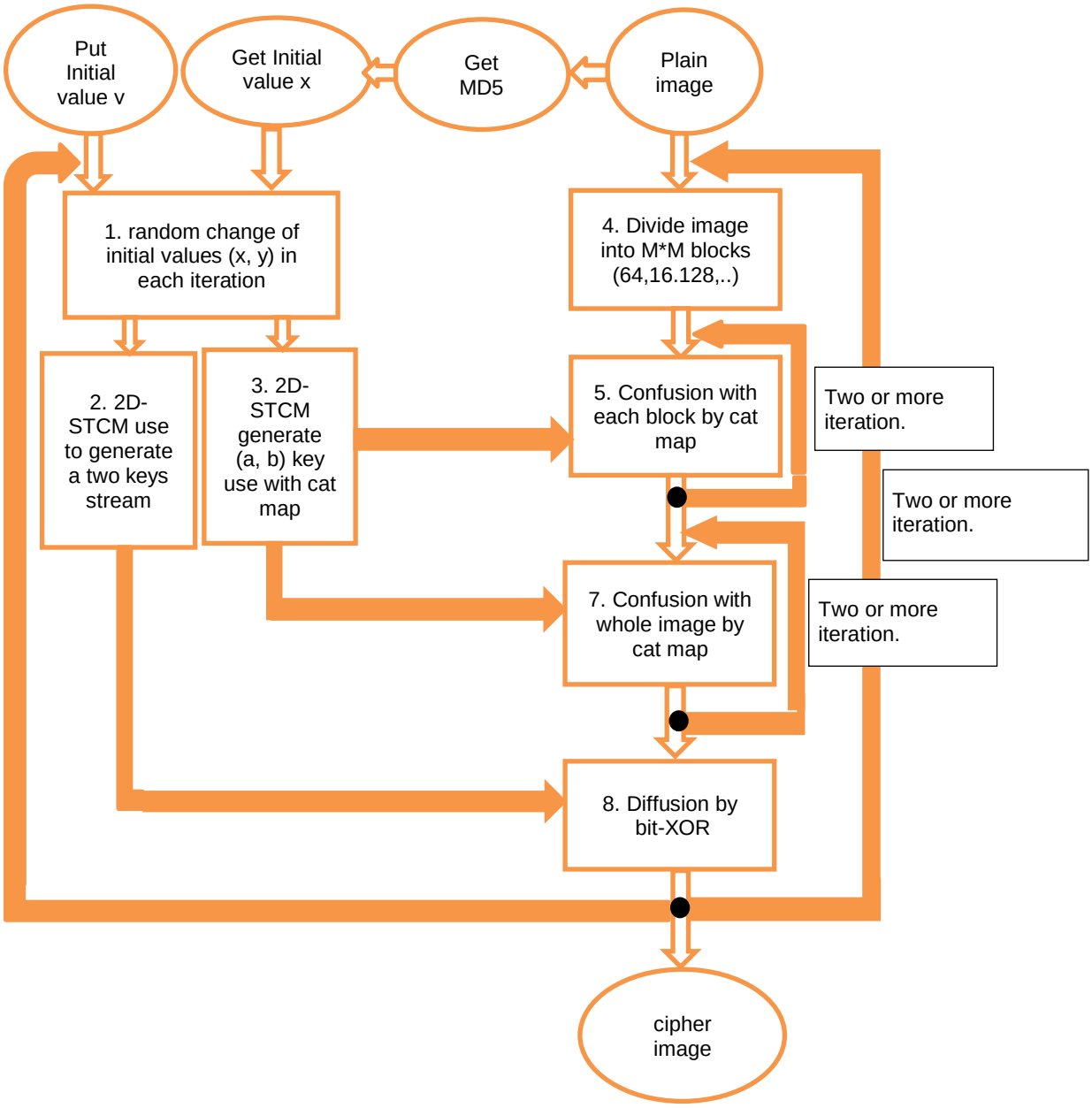

FIGURE 2: Block Diagram of The Proposed Encryption Scheme.

4. EXPERIMENTAL RESULT

The algorithm proposal encrypts the original image through confusion and diffusion using multiple steps. The algorithm is implemented using the MATLAB 2015a simulation program, several gray images from University of Southern California - Signal and Image Processing Institute (USC-SIPI) and system configuration Intel® core™ i3 CPU M330- @2.13 GHz and 4 GB RAM, operating system 64 bits. A comparison is done with the many other algorithms in [16, 17, 18, 19, 20, 21]. The results showed that our algorithm performs better or similar than other algorithms.

4.1 Key Space

The key space of encryption algorithm should be large enough to resist brute-force attack. Diffusion Key has five parameters ($x. y. \Delta x. \Delta y. n$); four parameters ($x, y, \Delta x, \Delta y$) represented by a double data type, each parameter is 64 bits and parameter (n) is real number represented by

8 bits. Diffusion key space size in this paper is $2^{264}$and it is large enough to resist brute-force attack.
In addition, there are confusion keys in this paper represented by key (a, b) of cat map, number of internal iterations and block size change, they are longer than 2^{40}. Total key space is longer than 2^{304}.

4.2 Key Sensitivity

A small change in security key causes large change in cipher image and a small change in security key can't be recover an original image as show in figures below.
Encryption Key 1: x_0=0.0058052, y_0=0.0352572, Δx = 0.01, Δy = 0.01, n=5.
Decryption Key 2: x_0=0.0058052, y_0=0.0352571, Δx = 0.01, Δy = 0.01, n=5.

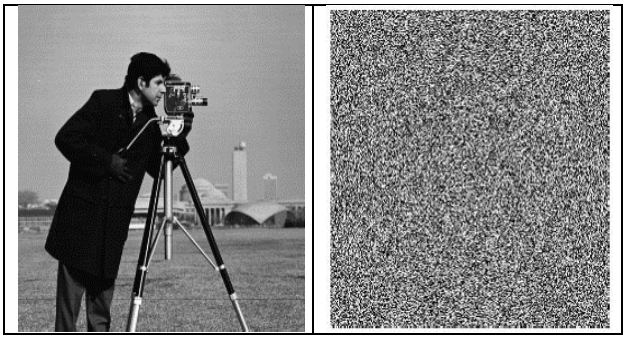

FIGURE 3: An Original Image Encryption by Key 1 and Decryption by Key 2.

4.3 Histogram

The histogram plots the occurrences frequency of image pixel value that show statistical features of images. Y-axis represents number of image pixels and X-axis represents intensity level. The histogram of plain image is not distributed uniformly, this means statistically pixels can be predicted, but histogram of cipher image is distributed uniformly, this means statistically pixels can't be predicted. Analyzed histograms of three plain images and their ciphered images. Figure 4 gives the plain images of 'Cameraman', 'Lena', 'Couple', and their histograms, respectively. Figure 5 gives the ciphered images of 'Cameraman', 'Lena', 'Couple', and their histograms respectively. The results shown in Figures 4 and 5 [images from USC-SIPI] indicate that our algorithm resists statistical attack.

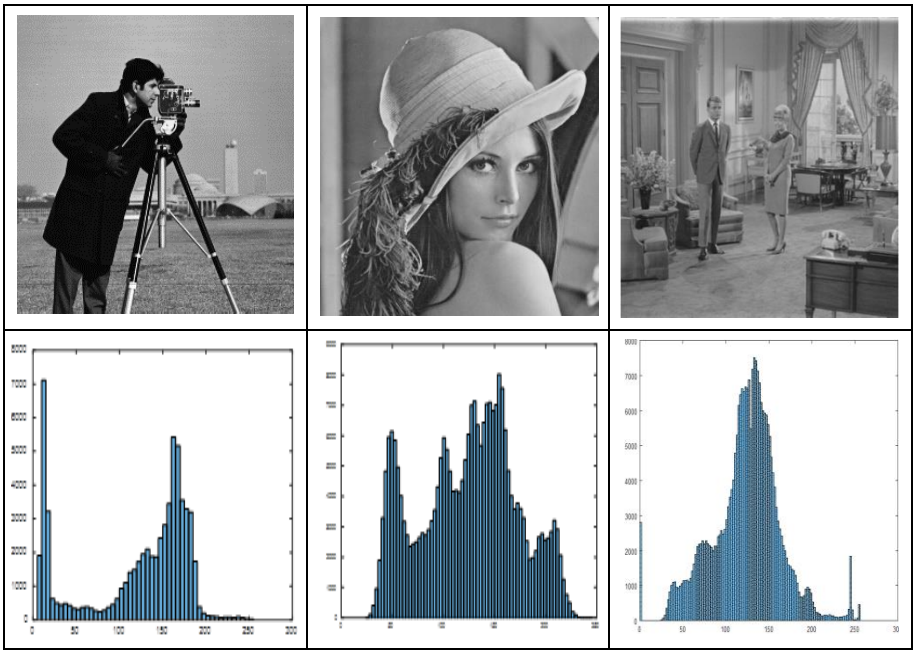

FIGURE 4: Plain images of 'Cameraman', 'Lena', 'Couple', and their histograms respectively.

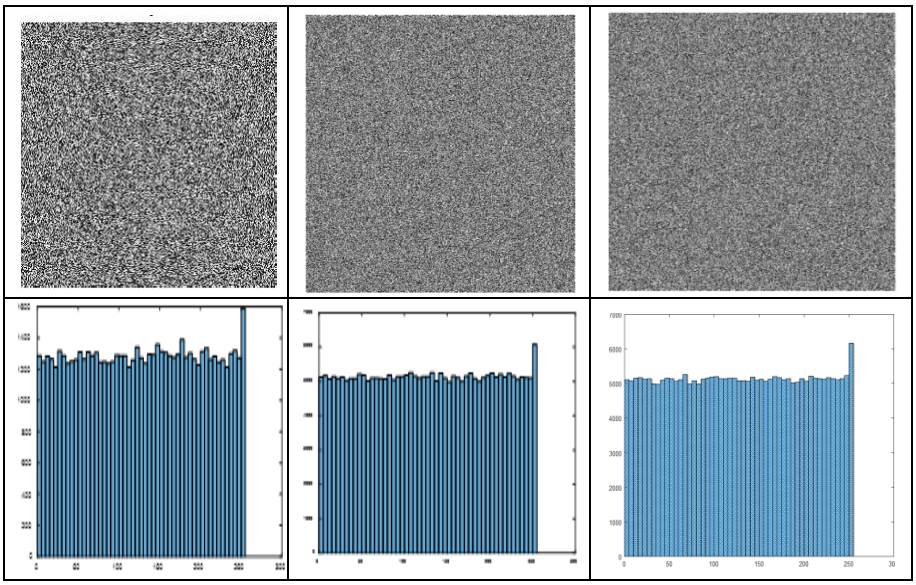

FIGURE 5: Ciphered images of 'Cameraman', 'Lena', 'Couple', and their histograms respectively.

4.4 Pixel Correlation

This analysis is used to study the correlation between two adjacent pixels in a plain image and a cipher image. In the plain image, the correlation between adjacent pixels are high, but in the cipher image should be low correlation between the adjacent pixels to resist statistical attacks. The below equations calculate the correlation coefficient as follows:

$$E_x = \frac{1}{N}\sum_{i=1}^{n} x_i \quad . \tag{11}$$

$$D_x = \frac{1}{N}\sum_{i=1}^{n}(x_i - E_X)^2 \ . \tag{12}$$

$$\mathrm{cov}(x, y) = \frac{1}{N}\sum_{i=1}^{n}(x_i - E_X)(y_i - E_y) \quad . \tag{13}$$

$$r_{xy} = \frac{\mathrm{cov}(x,y)}{\sqrt{D_x}\sqrt{D_y}} \quad . \tag{14}$$

In the above equations, in equation (11) calculates mean, in equation (12) calculates stander division, in equation (13) calculates covariance and in equation (14) calculates correlation coefficient**.** In below Table 1 illustrates correlation coefficient values of the plain image and its cipher image.

4.5 Information Entropy

Entropy is a statistical measure that use to randomness test in the plain image and ciphered image. The information of entropy is given by equation (15):

$$H(X) = -\sum_{i=1}^{2^N} \Pr(x_i)\log_2 \Pr(x_i) \tag{15}$$

Here $H(X)$ is the entropy value, $\Pr(x_i)$ is the probability of symbol x_i, and N is the number of bits to represent a symbol x_i. When $H(X)$ = N this ideal entropy value. The results shown in Tables 2 indicate that our algorithm resists entropy attack.

4.6 NPCR and UACI Tests

NPCR (Number of pixels change rate) and UACI (Unified average changing intensity) are used in plain image sensitivity tests. NPCR is approaching to 100%, when changing one pixel in the plain image, resulting in a complete change to the cipher image. Let's take a look at this example. I_1 is a plain original image, we change one pixel to get another plain image I_2, the result after the encryption process is two completely different C_1 and C_2 ciphered images.

Arithmetically, the value of the bipolar array $D_{(i,j)}$ can be calculated as follows:

$$D_{(i.j)} = \begin{cases} 0. & \text{if } C_{1_{(i.j)}} = C_{2_{(i.j)}} \\ 1. & \text{if } C_{1_{(i.j)}} \neq C_{2_{(i.j)}} \end{cases} \quad (16)$$

Then NPCR can be calculated as follows:

$$NPCR = \frac{1}{N \times M} \sum_{i=1}^{M} \sum_{j=1}^{N} D_{(i.j)} \times 100\% \quad (17)$$

plain image sensitivity is better when the NPCR value is close to 100%. UACI measures the average intensity between the C_1 and C_2 ciphered images, obtained from plain images I_1 and I_2. It can be defined as follows:

$$UACI = \frac{1}{N \times M} \left[\sum_{i=1}^{M} \sum_{j=1}^{N} \frac{C_{1(i.j)} - C_{2(i.j)}}{255} \right] \times 100\% \quad (18)$$

plain image sensitivity is better when the UACI value is close to 33%. The results shown in Tables 3.4 indicate that our algorithm resists differential attack.

4.7 Comparison of Various Image Encryption Techniques

Comparison is done on the basis of; direction, plain image and encrypted image of our scheme with various image encryption algorithms

Image	Direction	Plain image	Encrypted image						
			Our scheme	Ref. [16]	Ref. [17]	Ref. [18]	Ref. [19]	Ref. [20]	Ref. [21]
Airplane	Horizontal	0.957	0.0075	0.0033	−0.0002	0.0011	−0.0164	0.0004	0.0045
	Vertical	0.9365	-0.001	0.0002	−0.0090	−0.0035	−0.0181	0.0061	−0.0215
	Diagonal	0.8926	0.0037	−0.0019	−0.0066	−0.0029	0.0004	−0.0079	−0.0015
Airport1	Horizontal	0.9099	-0.0018	−0.0021	−0.0275	0.004	−0.0061	0.0094	−0.0142
	Vertical	0.9033	0.0008	0.0002	−0.0154	−0.0174	−0.0079	−0.0107	−0.0141
	Diagonal	0.859	-0.0006	0.0003	−0.0187	−0.0135	−0.0001	−0.0007	0.0002
Barbara	Horizontal	0.8953	0.0005	0.0033	−0.0033	−0.0052	−0.0212	−0.0187	0.0037
	Vertical	0.9588	-0.0004	0.0032	−0.0269	−0.0067	−0.0161	−0.0016	−0.0202
	Diagonal	0.883	0.0015	0.0025	−0.0121	0.0068	−0.0110	0.0001	0.0046
Boat	Horizontal	0.9381	0.0005	0.0025	−0.0100	−0.0054	−0.0189	−0.0295	−0.0138
	Vertical	0.9713	-0.0026	0.0018	−0.0124	−0.0009	0.0003	−0.0150	−0.0199
	Diagonal	0.9221	0.0022	0.0005	−0.0185	0.0026	−0.0204	−0.0224	−0.0057
Camera man	Horizontal	0.9334	0.0035	0.0047	−0.0095	−0.0211	0.0063	−0.0047	−0.0009
	Vertical	0.9592	-0.0002	−0.0054	−0.0170	−0.0103	−0.0142	−0.0195	−0.0223
	Diagonal	0.9086	-0.0025	0.0016	−0.0119	0.0054	0.0168	0.0279	0.0025
Chemical plant	Horizontal	0.9466	-0.0014	−0.0030	−0.0134	−0.0073	−0.0069	−0.0091	0.0072
	Vertical	0.8984	0.0025	−0.0019	−0.0005	−0.0073	−0.0100	−0.0029	−0.0015
	Diagonal	0.8529	-0.004	−0.0063	−0.0033	−0.0115	−0.0078	−0.0092	−0.0040
Clock	Horizontal	0.9564	0.0078	0.0008	0.0024	−0.0140	−0.0248	−0.0143	−0.0123
	Vertical	0.974	0.0059	0.0013	−0.0246	−0.0139	−0.0172	0.0097	−0.0041
	Diagonal	0.9389	0.0017	−0.009	−0.0081	−0.0175	−0.0025	0.012	−0.0027

Couple	Horizontal	0.937	0.0007	0.0012	−0.0251	−0.0178	−0.0122	−0.0236	−0.0106
	Vertical	0.8926	-0.0034	0.0011	−0.0213	−0.0025	0.0262	−0.0045	−0.0047
	Diagonal	0.8558	0.0022	0.0013	−0.0078	0.0001	−0.0257	0.0016	0.0262
Lena	Horizontal	0.9719	0.0009	−0.0047	−0.0048	−0.0086	−0.0066	0.0011	−0.0063
	Vertical	0.985	-0.0009	0.0015	−0.0112	−0.0102	−0.0089	0.0098	0.0098
	Diagonal	0.9593	-0.0027	0.003	−0.0045	−0.0125	0.0424	−0.0227	−0.0154
Man	Horizontal	0.9774	0.0004	−0.0010	−0.0155	−0.0190	−0.0083	0.0022	−0.0100
	Vertical	0.9812	0.00004	0.0029	−0.0276	−0.0095	−0.0180	−0.0226	−0.0027
	Diagonal	0.9671	-0.0019	−0.0005	−0.0157	−0.0141	0.025	0.006	−0.0195
Moon surface	Horizontal	0.902	0.0018	−0.0061	−0.0062	−0.0288	−0.0065	−0.0212	−0.0063
	Vertical	0.9389	0.0013	0.0001	−0.0075	−0.0194	0.0063	−0.0166	0.0142
	Diagonal	0.9037	0.0011	0.0021	−0.0044	−0.0137	0.0135	0.0138	−0.0091
Tank	Horizontal	0.9456	-0.0012	0.0033	−0.0202	0.0049	−0.0185	−0.0383	−0.0159
	Vertical	0.932	0.0012	0.0002	−0.0200	−0.0128	−0.0119	−0.0362	−0.0114
	Diagonal	0.9017	-0.0005	−0.0019	−0.0013	−0.0011	−0.0249	0.022	0.0126
Average	Horizontal		0.002	0.003	0.0115	0.0114	0.0127	0.0143	0.0088
	Vertical		0.0009	0.0016	0.0161	0.0095	0.029	0.0129	0.0122
	Diagonal		0.0026	0.0025	0.0094	0.0084	0.0158	0.0121	0.0086

TABLE 1: Comparing the correlation coefficient of our algorithm with other algorithms.

Image name	Our scheme	Ref. [16]	Ref. [17]	Ref. [18]	Ref. [19]	Ref. [20]	Ref. [21]
Airplane	7.997	7.9998	7.996	7.9966	7.9966	7.9969	7.9957
Airport	7.9998	7.9972	7.9974	7.9965	7.9926	7.9954	7.9966
Barbara	7.9992	7.9993	7.9978	7.9964	7.9937	7.9957	7.9964
Boat	7.9993	7.9993	7.998	7.9979	7.996	7.9959	7.9985
Camera man	7.9975	7.9972	7.9985	7.9966	7.9955	7.9964	7.999
Chemical plant	7.9971	7.9971	7.9964	7.9986	7.9947	7.999	7.994
Clock	7.9975	7.9975	7.9992	7.9974	7.9984	7.9956	7.9977
Couple	7.9991	7.9993	7.9976	7.9987	7.9951	7.998	7.9956
Lena	7.9993	7.9991	7.9963	7.9991	7.9951	7.9965	7.9964
Man	7.9998	7.9998	7.9975	7.9974	7.999	7.9965	7.9949
Moon surface	7.9972	7.9975	7.9987	7.9979	7.9951	7.9954	7.9969
Tan	7.9992	7.9994	7.9969	7.999	7.9976	7.9965	7.9997
Average	7.999	7.998542	7.997525	7.997675	7.995783333	7.99648333	7.99678333

TABLE 2: Comparing the information entropy of our algorithm with other algorithms.

Image name	Our scheme	Ref. [16]	Ref. [17]	Ref. [18]	Ref. [19]	Ref. [20]	Ref. [21]
Airplane	99.62	99.5849	99.615	99.6107	99.5662	99.618	99.6201
Airport	99.5984	99.6459	99.6083	99.6077	99.5565	99.6231	99.5499
Barbara	99.6307	99.6215	99.6092	99.6162	99.5227	99.6402	99.5178
Boat	99.6074	99.6227	99.6102	99.6281	99.5609	99.6209	99.5743
Camera man	99.6154	99.6124	99.6205	99.6292	99.5749	99.6105	99.6216
Chemical plant	99.617	99.62	99.6121	99.6131	99.5325	99.6258	99.6185
Clock	99.62	99.5727	99.6102	99.6218	99.591	99.6126	99.5728
Couple	99.612	99.6185	99.6399	99.6292	99.5606	99.6312	99.5468
Lena	99.5914	99.6253	99.6228	99.6146	99.5511	99.6092	99.6231
Man	99.6057	99.6189	99.607	99.6084	99.5417	99.6211	99.5514
Moon surface	99.6429	99.6276	99.6139	99.6168	99.5684	99.6172	99.6033
Tank	99.6059	99.6307	99.629	99.6131	99.5794	99.6687	99.6078
Average	99.6139	99.61676	99.61650833	99.6174083	99.55882	99.624875	99.58395

TABLE 3: Comparing the NPCR of our algorithm with other algorithms.

Image name	Our scheme	Ref. [16]	Ref. [17]	Ref. [18]	Ref. [19]	Ref. [20]	Ref. [21]
Airplane	33.4188	33.6829	33.5625	33.6632	33.3716	33.6183	33.5961
Airport	33.4354	33.4188	33.5359	33.4143	33.4063	33.5817	33.664
Barbara	33.4914	33.4691	33.7431	33.5776	33.389	33.6714	33.5052
Boat	33.4665	33.5137	33.5367	33.6145	33.4176	33.6018	33.3975
Camera man	33.4165	33.6551	33.7786	33.705	33.3691	33.6862	33.7326
Chemical plant	33.3856	33.4291	33.6068	33.8543	33.3872	33.3825	33.5831
Clock	33.4893	33.4444	33.582	33.4836	33.4147	33.5793	33.9272
Couple	33.522	33.3615	33.8085	33.6446	33.3723	33.7252	33.8911
Lena	33.4699	33.4807	33.7041	33.5561	33.3461	33.6322	33.8144
Man	33.4834	33.4815	33.7905	33.6744	33.3684	33.472	33.6949
Moon surface	33.2323	33.4408	33.6898	33.5333	33.3758	33.6993	33.8063
Tank	33.4714	33.5212	33.9213	33.7155	33.4045	33.5224	33.7415
Average	33.4402	33.49157	33.68831	33.6197	33.3852	33.59769	33.6961

TABLE 4: Comparing the UACI of Our Algorithm with Other Algorithms.

5. CONCLUSION AND FUTURE WORK

This paper supports a set of confusion and diffusion methods, and avoids poor key management processes to achieve high security. The process of confusion deals with changing the position of the pixel by applying a chaotic map (the cat). The diffusion process deals with changing the pixel value by performing a bit-XOR operation between the 2D-STCM and the confused image. In the algorithm, confusion and diffusion pass through two or more iterations, in each iteration new key sequences are created by 2D-STCM. Therefore, the total key space is long enough to resist brute-force attack and increase security level.

From experimental results the key space analysis, key sensitivity analysis, histogram, correlation coefficient, information entropy, NPCR and UACI show very good result. As a result, the encryption algorithm is resistant to attacks such as statistical and differential attacks. The cipher image has a high security level not only because of confusion and diffusion methods, but also any key should only be used once. After comparison it is found that in our paper correlation coefficient was reduced and reach 29% fewer than the results of similar references in contrast, entropy information is higher than the listed papers (tends to 8) and NPCR is similar or higher than the listed papers. The obtained results show that the submitted image information becomes random. Future work will deal with color encryption schemes using different methods of image fusion.

6. REFERENCES

[1] F. Nasser and M. Mohamed. “Improvement of Images Encryption Based on Multiple Chaotic Maps and 2D-STCM Key Space,” presented at the 3rd Int. Conf. Computer Science and Engineering Information technology, Aden, Yemen, 2019.

[2] W. Stallings. Cryptography and Network Security Principles and Practice, 4th ed. USA: Prentice Hall, 2006, pp.1-210.

[3] B. Schneier. Applied Cryptography, Protocols, Algorithms, and Source Code, second ed. New York: John Wiley & Sons, Inc., 1996, pp. 370-642.

[4] S. Lian. “A Block Cipher Based on Chaotic Neural Networks.” Elsevier, Neuro computing, vol. 72, pp. 1296-1301, Nov. 2009.

[5] M. Billal Hossain, M. Toufikur Rahman, S. Rahman and S. Islam. "A New Approach of Image Encryption Using 3D Chaotic Map to Enhance Security of Multimedia Component," presented at the 3rd Int. Conf. Communication Engineering Department, Bangladesh, 2014.

[6] M. Mishra, P. Singh and C. Garg. “A New Algorithm of Encryption and Decryption of Images Using Chaotic Mapping.” International Journal of Information and Computation technology, Vol. 4, pp. 741-746, 2014.

[7] C.E. Shannon. Communication theory of secrecy systems. Bell. Syst. Tech. 1949, pp. 656-715.

[8] K. Naik and A. Kumar Pal. "An Image Cryptosystem based on Diffusion of Significant Bit-planes of a Scrambled Image with Generated Binary Key Matrices," Proceedings of IEEE (ICCICR), 2013, pp. 15.

[9] S. Som and A. Kotal. "Confusion and Diffusion of Grayscale Images Using Multiple Chaotic Maps," Proceedings of IEEE (NCCCS), 2012, pp. 15.

[10] Z. Lv, L. Zhang, and J. Guo. “A Symmetric Image Encryption Scheme Based on Composite Chaotic Dispersed Dynamics System,” in Proc. ISCSCT, 2009, pp.191-194.

[11] S. Agarwal. “Secure Image Transmission Using Fractal and 2D-Chaotic Map.” The International Journal of Imaging, pp. 17, Jan. 2018.

[12] L. Hongjun and W. Xingyuan. "Color Image Encryption based on One-Time Keys and Robust Chaotic Maps." The International Journal of Computers and Mathematics with Applications, pp. 3320–3327, Mar. 2010.

[13] S. Agarwal, G. Srivastava and A. Negi. " Dynamics of Mandelbrot set with transcendental function." The International Journal of Advanced Computer Science and Applications, Vol. 3, No.5, pp. 142-146, 2012.

[14] X. Tong, Y. Liu, M. Zhang, H. Xu and Z. Wang. “Image Encryption Based on Hyperchaotic Liu System Algorithm.” The International Journal of Eng. & Tech. Journal, Vol.33, No.17, pp.181-196, Jan. 2015.

[15] R. Rivest. “MD5.” Internet: https://en.wikipedia.org/wiki/MD5, May. 29, 2019.

[16] R. Zahmou, R. Ejbali and M. Zaied. “Image Encryption Based on New Beta Chaotic Maps.” The International Journal of Opt. Lasers Eng., Vol.96, pp. 39–49, Apr. 2017.

[17] A. Belazi, A. Abd El-Latif and S. Belghith, “A Novel Image Encryption Scheme Based on Substitution-Permutation Network and Chaos.” The International Journal of Signal Process pp.41, Mar. 2016.

[18] X. Wang, L. Teng and X. Qin. “A Novel Colour Image Encryption Algorithm Based on Chaos.” The International Journal of Signal Process, pp.1101–1108, Oct. 2011.

[19] X. Wang, L. Liu and Y. Zhang. “A Novel Chaotic Block Image Encryption Algorithm Based on Dynamic Random growth Technique.” The International Journal of Opt. Lasers Eng., Vol.66, pp.10–18. Aug. 2014.

[20] Z. Hua, Y. Zhou, C. Pun and C.L. Philip Chen. “2D Sine Logistic Modulation Map for Image Encryption.” The International Journal of Inf. Sci. Vol.297, pp.80–94, Nov, 2015.

[21] XY. Wang, L. Yang, R. Liu and A. Kadir. “A Chaotic Image Encryption Algorithm Based on Perceptron Model.” The International Journal of Nonlinear Dyn, pp. 615–621, Jun, 2010.

[22] G. Ye* and X. Huang. "A secure Image Encryption Algorithm Based on Chaotic Maps and SHA-3." Security and Communication Networks, pp. 2015–2023, Feb. 2016.