A Mixed Chaotic-cellular Automata Based Encryption Scheme for Compressed Jpeg Images

Rabab Beniani
UDL-University
Algeria
rabab-sba@hotmail.fr

Kamel Mohamed FARAOUN
UDL-SBA
Algeria
kamel_mh@yahoo.fr

ABSTRACT: *In this paper, we propose a new scheme for joint compression and encryption of digital images, based on cellular automata and selective encryption of quantized DCT coefficient. Using a key stream by a cellular automata mechanism, a subset of quantized coefficients is selected and then ciphered with the produced pseudorandom key-stream. Thus, we achieve a sufficiently robust security level, mostly against known plaintext attacks while preserving a high compression ratio. Several performance analysis including security analysis, speed performances and compression ratio are performed, and demonstrate the efficacy of the proposed approach with respect to existing ones.*

Keywords: Selective Images Encryption, Jpeg Images, Cellular Automata, Chaotic Maps

DOI: 10.6025/jmpt/2018/9/3/88-101

Received: 4 March 2018, Revised 29 April 2018, Accepted 7 May 2018

1. Introduction

In recent years, proliferation of smart phones and cameras on the one hand, and the change of the modes of human communication on the other (facebook, Twitter, Instagram...), have facilitated the realization and sending multimedia content. Such a development puts us the need to reduce the size of the transmitted/stocked data, and more precisely images and videos that seems to be the most generated, to provide faster data transmission across the communication lines and at the same time ensured their protection against ears indiscreet. That's why, the need for compression and encryption is growing, and both compression and encryption have received increasing attention from researchers in the field of information security. So far, a large number of papers focusing on image compression [1-4] and image encryption [5-6] have been reported.

Unfortunately, the realization of these two cascading operations induces a certain number of problems. Indeed, compressing data and encrypting it subsequently or vice versa can reduce the compression ratio as well as the robustness of the encryption. To solve this problem, several combination approaches have been proposed. There exist two separate research axes for joint compression and encryption. The first includes the compression in cryptographic algorithms. For example, Wong and Yuen proposed an algorithm for integrating compression into the chaotic cryptosystem type Baptista [7]. However, the approach has suffered a reduction in compression performance since the encryption operation reduces the correlation of the original data significantly.

The second axis of research is based on introducing encryption during compression steps. What is interesting in this type of encryption is that it retains the original functionality of the compression standard and preserves data compatibility. This is generally done by direct manipulation of the entities found in the data flow of the compressed content as the DCT coefficients [8] the quantized coefficients [9,10] the Huffman trees [11] the modification of the scanning order [13] permutation or masking of the Huffman code words [14,15,16]. However, most of these encryption methods are not completely secure [12, 17].

On the other hand, cellular automata CA demonstrate excellent means of simulating the complicated and pseudo-random behaviours that are essential for efficient ciphers. A number of research work on image ciphering based on cellular automata has been performed [18, 19, 20]. A model based on DCT (Discrete Cosine Transform) and one dimensional cellular automata was described in [18]. Chen et al. proposed an encryption and compression scheme based on Elementary Cellular Automata and Kronecker Compressed Sensing (KCS) [19]. The testing shows that Image encryption and compression based on the application of cellular automata method gives a higher level of confidentiality. This paper proposes an encryption scheme of JPEG compressed images. Only a subset of quantified coefficients are selected in a random way using cellular automata keystream and then encrypted using keystream to achieve a high sensibility to plaintext variations and then resist to the know plaintext attack. The aim of the proposed approach is to ensure a high randomness quality of the ciphered JPEG images, while preserving a sufficiently high compression ratio, in contrast to existing approaches that produce low random compressed images leading to possible known plain text attacks.

The rest of this paper is as follows: In the next section, a brief introduction of cellular automata and JPEG compression is presented. In Section 3, we introduce the proposed scheme in detail. In section 4, security of the schema is analyzed and performance evaluations are demonstrated eventually. Finally, conclusions are drawn in Section 5.

2. Theoretical Preliminaries

2.1. JPEG and Discrete Cosine Transform

Based on discrete cosine transform, jpeg is very powerful to give a minimal size to an image, this compression process comports five main steps as is shown in Figure 1.

In JPEG encoding, the RGB image is transformed into YCbCr colour space, which will be subsequently divided into 8x8 pixel blocks. Each 8x8 block of each component (ycbcr) are further transformed into DCT domain. The DCT works by separating images into parts of differing frequencies by using the following formula (1):

$$F_k(u, v) = \frac{c(u)\, v(v)}{4} \Sigma_{i=0}^{7} \Sigma_{i=0}^{7} fk(i, j) \cos\left[\frac{(2i+1)u\pi}{16}\right] \cos\left[\frac{(2i+1)v\pi}{16}\right] \quad (1)$$

$$c(e) = \begin{cases} 1/\sqrt{2} & \text{if } e = 0 \\ 1 & \text{if } e \neq 0 \end{cases}$$

Where $f(i,j)$ is the pixel values at the position (i,j) and $f(u,v)$ is the DCT coefficient at coordinate (u,v) of block K. Then each of the 8x8 DCT coefficient of the colour space y and cb, cr are quantized by the luminance and chrominance quantization table respectively. The elements in the quantization matrix control the compression ratio, with larger values producing greater compression. A typical quantization matrix (for a quality of 50% as specified in the original JPEG Standard), is shown in table1.

After quantization the AC coefficients are scanned using zigzag-scan as shown in Figure 2 than the replications of the zeros are exploited by an RLE (run length encoding). The coefficients DC are coded by differential pulse code modulation DPCM, and

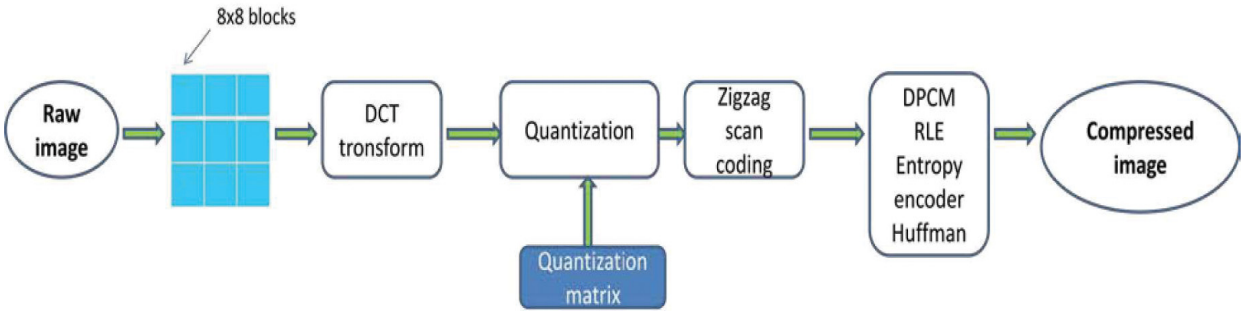


Figure 1. JPEG compression process

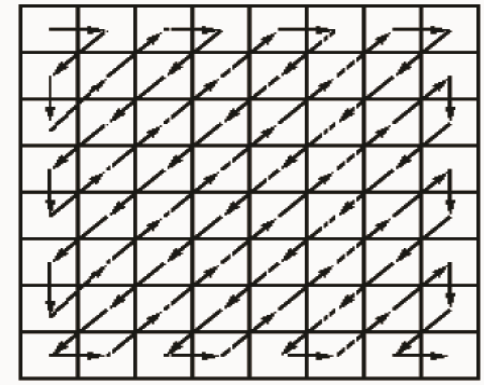


Figure 2. Zigzag scan illustration

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

(a)

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

(b)

Table 1. The quantization tables (a) Luminance, (b) Chrominance

finally the symbols obtained by the application of the two preceding coding are coded by entropy coding of Huffman.

In contrast to uncompressed images forms, JPEG images content is very specific and highly sensitive to alterations and modifications of the quantized coefficients. As a results, the task of encrypting JPEG images is completely different and more content-specific. Selective encryption is generally employed to encipher only a subset of coefficients, in order to achieve both efficient compression and robust encryption against cryptographic attacks. The following sections lists some encryption techniques that are employed to achieve selective JPEG encryption, especially those used by the proposed scheme.

2.2. Cellular Automata for Encryption

Cellular Automata are dynamical systems, discrete in both time and space. It consists of an infinite, regular grid of cells, each in one of a finite number of states. The grid can be in any finite number of dimensions. Time is also discrete and the state of a cell at time t+1 is a function of the states of a finite number of cells (called its neighborhood) at time t. These neighbors are a selection

of cells relative to the specified cell, and do not change.

The cellular array (grid) is *d*-dimensional, where $d = 1,2,3,..$ is used in practice; in this paper we shall concentrate on $d = 1$, i.e., one-dimensional grids. The identical rule contained in each cell is essentially a finite state machine, usually specified in the form of a rule table (also known as the transition function *f*), with an entry for every possible neighborhood configuration of states. The cellular neighborhood of a cell consists of itself and the surrounding (adjacent) cells. For one-dimensional *CA*s, a cell is connected to *r* local neighbors (cells) on either side where *r* is referred to as the radius (thus, each cell has $2r + 1$ neighbor). If Q is the states set, and the state of a cell I at a time t is x^t_i then the function f is a mapping:

$$f: Q \text{ x } Q \text{ x } ... \text{ x } Q \rightarrow Q$$

$$x_i^{t+1} = f(x^t_{i-r}, x^t_{i-r+1}, \ldots x^t_{i-1}, x^t_i, x^t_{i+1}, \ldots..., x^t_{i+r-1}, x^t_{i+r}) \quad (2)$$

Cellular automatons (CAs) are used for cryptographic purpose for both symmetric cryptosystems and public keys cryptosystems. The first proposed approach using CAs has been proposed by Wolfram [3] using the Rule 30 as random number generator. Hortensius et al. [21] and Nandi et al. [22] used non-uniform CAs with two rules 90 and 150, and it was found that the quality of generated pseudo random numbers was better than the quality of the Wolfram system. Recently Tomassini and Perrenoud [23] proposed to use non-uniform 1D CAs with $r = 1$ and four rules 90, 105, 150 and 165, which provide high quality PNSs and a huge space of possible secret keys which is difficult for cryptanalysis. Instead to design rules for CAs they used evolutionary technique called cellular programming (CP) to search for them. In [24], Gutowitz proposed a block based cryptosystem using reversible and irreversible CAs to produce the cipher text. Recent works use also 2-dimensional CAs [25,26, 27]. Most investigations into CA-based cryptosystems have been aimed at traditional secret key systems [28, 29]. There appear to be very few CA-based public key cryptosystems in the literature; one is the Finite Automata Public-Key Cryptosystem, or Tao-Chen cryptosystem [30], and there was another attempt by Guan [31] although they both use non-uniform CA. Kari's paper [32] outlines an idea for a public-key cryptosystem based on reversible cellular automata, and poses the question of how to implement the key generation algorithm.

2.3. Chaotic Maps for Random Numbers Generation

Chaos is a nonlinear phenomenon existing in deterministic nonlinear systems that exhibit extreme sensitivity to initial conditions and have random-like behaviors. Utilizing chaos as a chaotic sequence generator has become an important and exciting study field in the past decade, since it is non-periodic, non-convergent and extremely sensitive to the initial condition. Among the various nonlinear chaotic maps, the most famous and widely used, is the so-called logistic map, which is one of the simplest systems exhibiting order-to-chaos transitions, the basic logistic map is formulated as:

$$x_{i+1} = \mu . x_i . (1 - x_i) \quad (3)$$

Where $k = 1,2,\ldots.$, and $0 \leq \mu \leq 4$, μ is called the branch parameter. All the statistical properties show that the mixing property of the chaotic sequence equals to the effect of adding discrete white noise [33], when the μ parameter belong to [3.57,4]. This critical issue is what to be needed in the process of random numbers generation.

In the present work, we propose to generate pseudorandom sequences used to encrypt the image using a mixed approach between cellular automata and chaotic maps. In order to enhance the randomness quality of CA-based generated sequences, a chaotic selection of the lattice position from which the bit is selected at each CA iteration is generated using a discredited chaotic map according to a deterministic process. The following section details the proposed key stream generation scheme and the way it is integrated into the JPEG images encryption model.

3. Proposed Approach

3.1. Generation of the Keystream :a mixed CA-chaos Approach

In classical one-dimensional CA's bit key stream generators, the selected bits are collected with respect to a fixed site position of the lattice, that is maintained during all iterations. While the obtained sequence seem pseudo random, its reconstruction is be easily performed due to the locality of the elementary rules (as exploited by Meier's attack). In order to further decorrelate the sequence and enhance its randomness, bit sequences *time spacing* and *site spacing* are proposed. Time spacing imply that not

all generated iterations are considered as part of the random sequence while site spacing imply changing position of the selected bit at new generated CA's row.

Generation of the steps for site and time spacing are performed using a chaotic one-dimensional non linear map to generate spacing and timing steps during the automate evolution. Chaotic maps are used to generate two independent pseudo random sequences S^{site} and S^{time} (for site and time spacing respectively). Elements of the sequence S^{site} belong to the set $\{1,\ldots,N\}$, when N is the number of possible site locations in the cellular automata space, equal to the size of the initial array of the CA, while the second sequence S^{time} that defines time spacing steps is constituted by a pseudo random real values belonging to [0,1], that permit de decide whether or not the bit at current iteration is considered or not with respect to a selection probability value P_{sel} (defined as a parameter of the system).

3.1.1. Site Spacing Sequence S^{site} Generation

Starting from an initial condition x_0, a chaotic map generate a chaotic orbit within a limited region $V = [x_{min}, x_{max}]$ called attractor of the chaotic map ([0,1] in the case of logistic map). In order to create a discrete sequence, we used a simple and efficient approach: first, a finite numerical orbit of length N is generated using the map (3) starting from an initial value x_0, when N is the number of sites, and x_0 is a real value derived from the key. The map's attractor V is then decomposed into N disjoined region $V_1,..V_N$, such that $V = \bigcup_{i=1}^{N} V_i$, and then association is created between the regions V_i and the numbers $1,2,\ldots N$ according to the usual ordering of the orbit $x_0, x_1,\ldots x_N$, so that each region V_i has an associated number N_{Vi} belonging to $\{1,\ldots N\}$. The association is randomly created according to the distribution of the orbit and so to initial condition x_0.

Starting from another value y_0, we can compute the orbit $y_0, y_1,\ldots$ using the same logistic map. According to the region V_i to which belong the value y_k, the value of the corresponding number N_{Vi} is considered as the output value of the sequence S^{site}. So we can generate an unlimited number of pseudo random values of the sequence S^{site} using the generated orbit from y_0.

As input to this procedure, we need only two real values x_0 and y_0, and we have the sequence S^{site} as output to be used next during the encryption process. The two values x_0 and y_0 are derived from the key of the cryptosystem.

3.1.2. Generation of the time spacing sequence S^{time}

Time spacing steps are generated according to a specific probability of selection. The sequence S^{time} is taken as the orbit of the logistic map generated from a starting point z_0. At each time step, the value z_i is compared to the probability P_{sel} and the bit of the current time step is taken if $z_i > P_{sel}$. Otherwise, if $z_i < P_{sel}$, the current iteration is skipped to the next one. The comparison is feasible since the z_i's will belong to [0, 1].

3.1.3. Key Stream Generation Process

The proposed cryptosystem will generate a pseudo random sequence of bit stream using the rule 30 CA, applied on an initial 1D array of bits used as the key. But instead of selecting bits directly from a fixed site location in each time step, a site and time spacing is introduced. The site spacing is handled using the elements of the sequence S^{time}, when the selection process (time spacing) is handled using the sequence S^{time}. The diagram of the figure.3 explains how bit key stream is generated in our approach.

Let L be desired key stream, K the key with size N, CA_i the cellular automata array at time i, and $S^{time}{}_i$, $S^{site}{}_i$ the two sequences of both site spacing and time spacing.

The initial key is used as initial state of the cellular automata, and also to generate the three real valued initial conditions x_0, y_0 and z_0. A key of size N is subdivided into three identical size bit strings of length (N div 3), that will give respectively the binary representation of x_0, y_0 and z_0. If the binary representation of N is $b_1 b_2 \ldots bN$ then we have:

$$x_0 = b_1 * 2^{-1} + b_2 * 2^{-2} + \ldots + b_{N \, div \, 3} * 2^{-(N \, div \, 3)}$$

$$y_0 = b_{(N \, div \, 3)+1} * 2^{-1} + b_{(N \, div \, 3)+2} * 2^{-2} + \ldots + b_{(2*N \, div \, 3)} * 2^{-(N \, div \, 3)}$$

$$z_0 = b_{(2*N \, div \, 3)+1} * 2^{-1} + b_{(2*N \, div \, 3)+2} * 2^{-2} + \ldots + b_N * 2^{-(N \, div \, 3)} \quad (4)$$

With a minimal key size of 128 bit, each real value will be coded on 42 bit that ensures a precision of 10^{-14}. The precision of the derived values is more accurate when using larger key size.

3.2. Description of the JPEG Encryption Scheme

A novel method for selective JPEG images encryption is proposed in the following. The aim is to achieve tow principal goals: resistance of the encryption scheme to the known plaintext attack and preservation of a sufficiently small size of data to ensure a good compression ration with best reconstruction quality. The diagram of the scheme is shown in Figure 4.

The proposed scheme includes the four following steps:

Step 1: After dividing the image into n 8x8 blocks, each on is transformed to the frequency domain using the DCT defined in equation (1) and then, the block is processed with quantization. When performing the zigzag-scan, a new scanning order is used to sweep the quantified elements, and hence complicate unauthorized reconstruction of the original image. Table 2 shows the default zigzag ordering and the new scanning ordering.

Step 2: Adopting the described approach described above, we generate a pseudo random sequence of bit stream, starting from the encryption key as initial 1D array of bits.

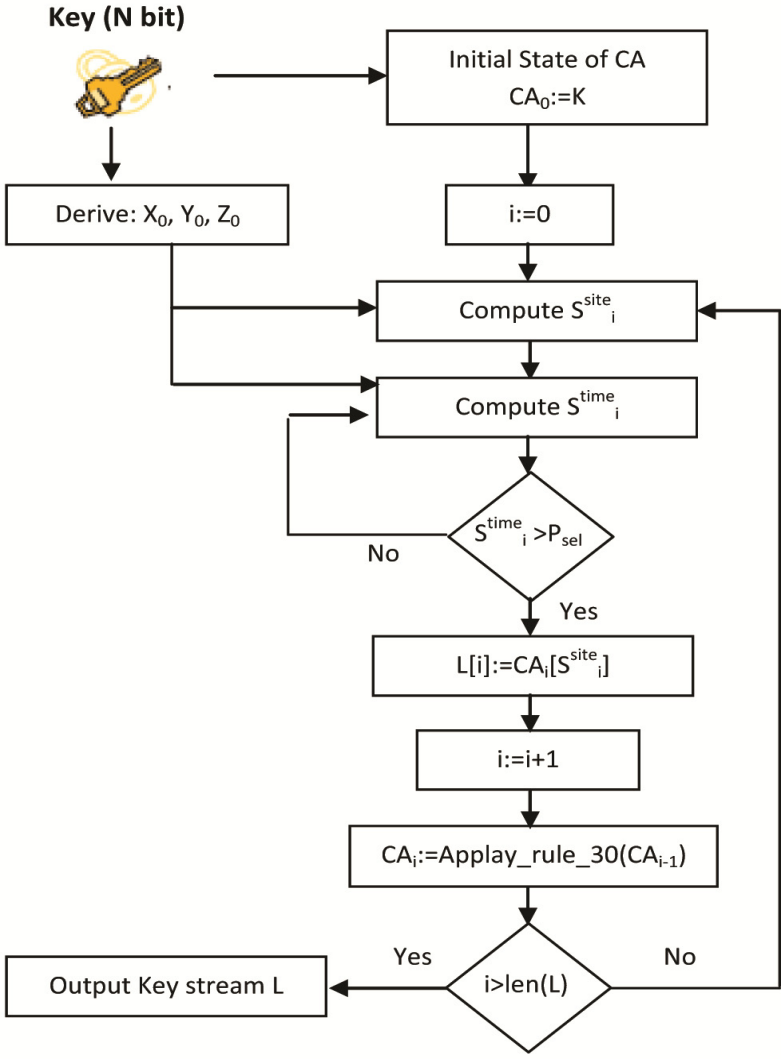


Figure 3. Diagram of the proposed key stream generation

Step 3: This step acts on each block of the DCT generated from the input image. A set of coefficients is first selected from each block according to the key stream and the selection threshold (±): The AC coefficients are chosen to be encrypted by taking those of the k^{th} position such that $k = L_i + \alpha$, where L_i is the i^{th} value in key stream. The coefficient selected will then be encrypted by a xor according to the Vernam model using the same key stream (of value L_i).

In order to achieve a high security level, we repeat the processing of a block several times, as for each iteration, a new value of key stream is applied. In this way a select element can be encrypted at least once or more with different key stream values.

Step 4: Encode the sequence obtained from step 3 in run-length coding and Huffman coding and complete compression-encryption of the plain image.

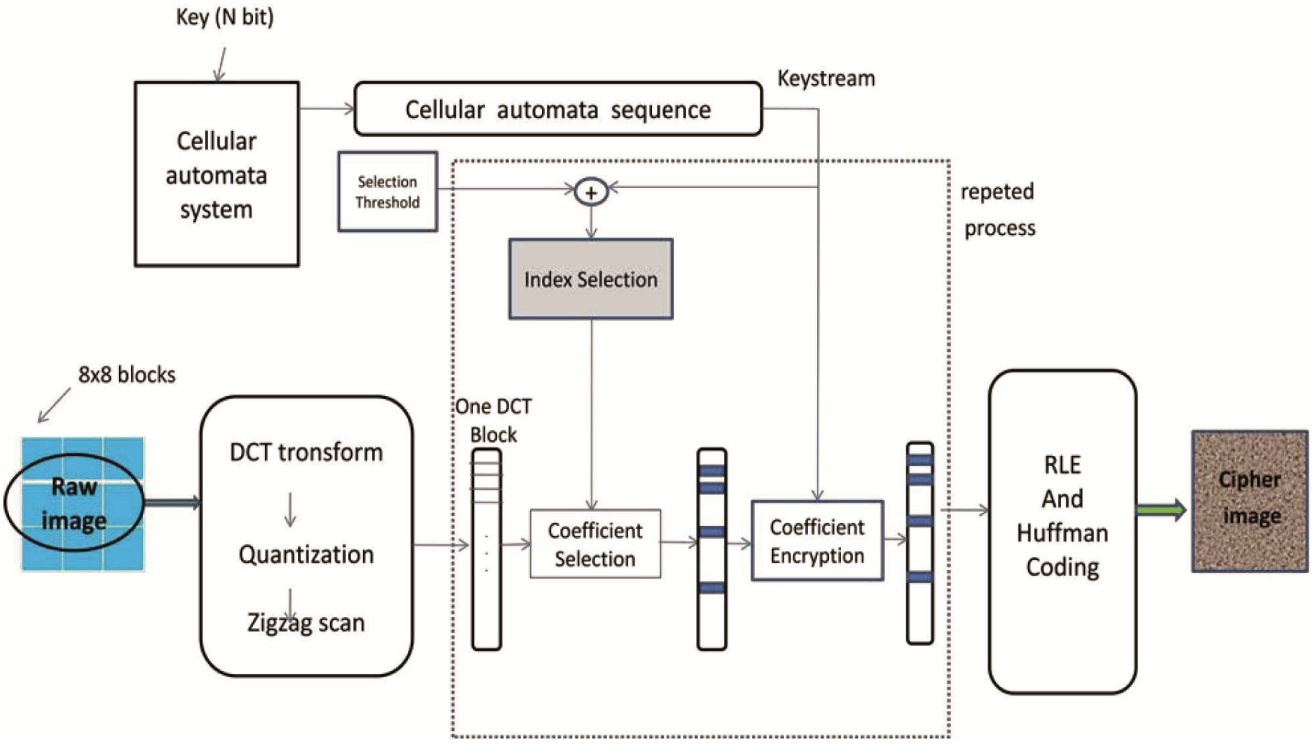


Figure 4. Proposed JPEG encryption scheme

1	3	4	10	11	21	22	36
2	5	9	12	20	23	35	37
6	8	13	19	24	34	38	49
7	14	18	25	33	39	48	50
15	17	26	32	40	47	51	58
16	27	31	41	46	52	57	59
28	30	42	45	53	56	60	63
29	43	44	54	55	61	62	64

(a)

1	2	6	7	15	16	28	29
3	5	8	14	17	27	30	43
4	9	13	18	26	31	42	44
10	12	19	25	32	41	45	54
11	20	24	33	40	46	53	55
21	23	34	39	47	52	56	61
22	35	38	48	51	57	60	62
36	37	49	50	58	59	63	64

(b)

Table 2. Default vs. proposed zigzag order : (a) default, (b) proposed order

4. Experimental Results

The proposed approach is implemented using Java Netbeans environment, while experiments are performed on an Intel Core i3 with2.4 GHz processor with 4GB memory. In the following, we show several performances evaluations of the proposed encryption and compression method with respect to reconstructed image quality, compression performance, speed performance, Key sensitivity analysis and correlation analysis.

4.1. Result of Encrypted and Reconstructed Image

The 512 × 512 Barbara gray image is encrypted and compressed using our proposed method to investigate encryption, compression and reconstruction quality. As an parameters, we set Quality level to $Q = 5, 25, 75$, respectively. The DCT coefficients of the original image are encrypted and compressed using our proposed method shown in Figure 5 (a-c). According to Figure 5(a-c) the encrypted and compressed images do not have any information of the original image. The reconstructed images are shown in Figure 6 (d-f). When Q = 5, the reconstructed images have many blurs. As CR increases, the reconstructed images have less blur. And when $Q = 75$, the reconstructed images are more similar with the original image. Therefore, the proposed method has a better encryption and compression performance.

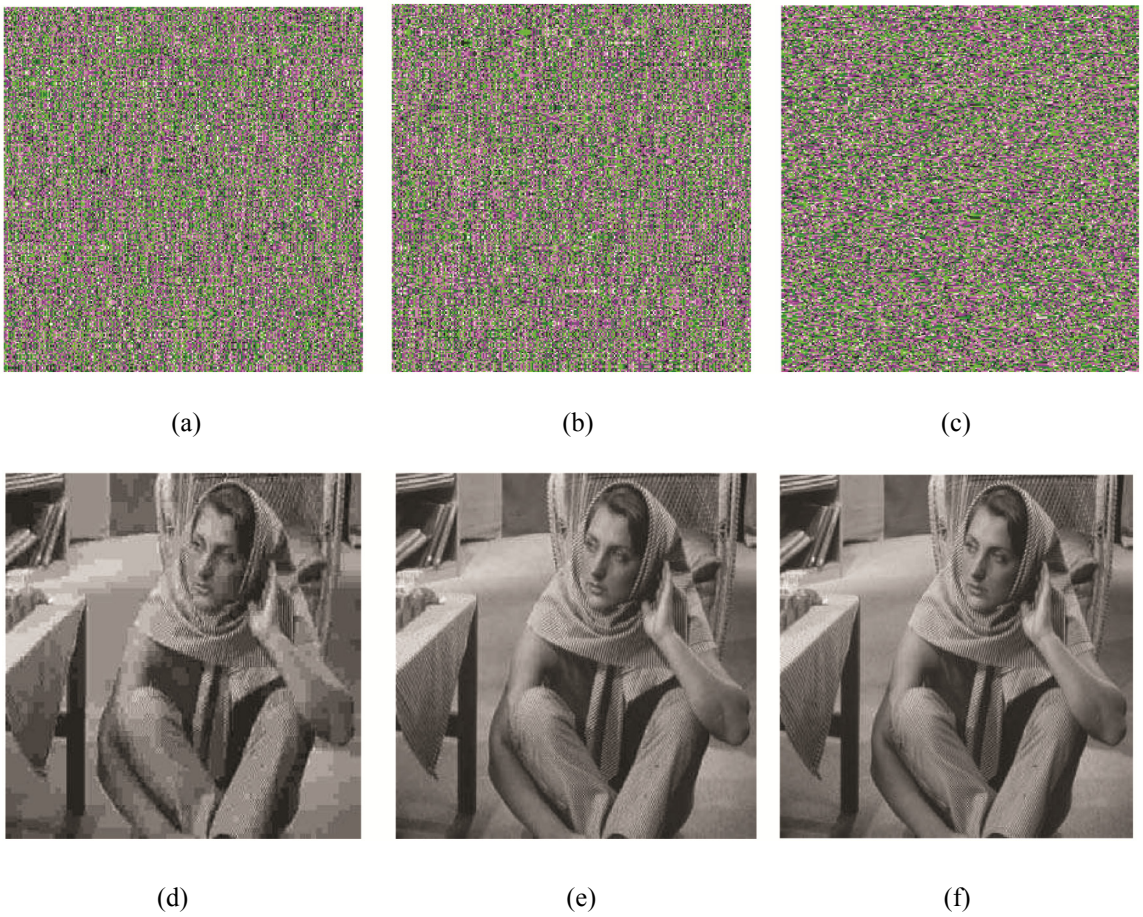
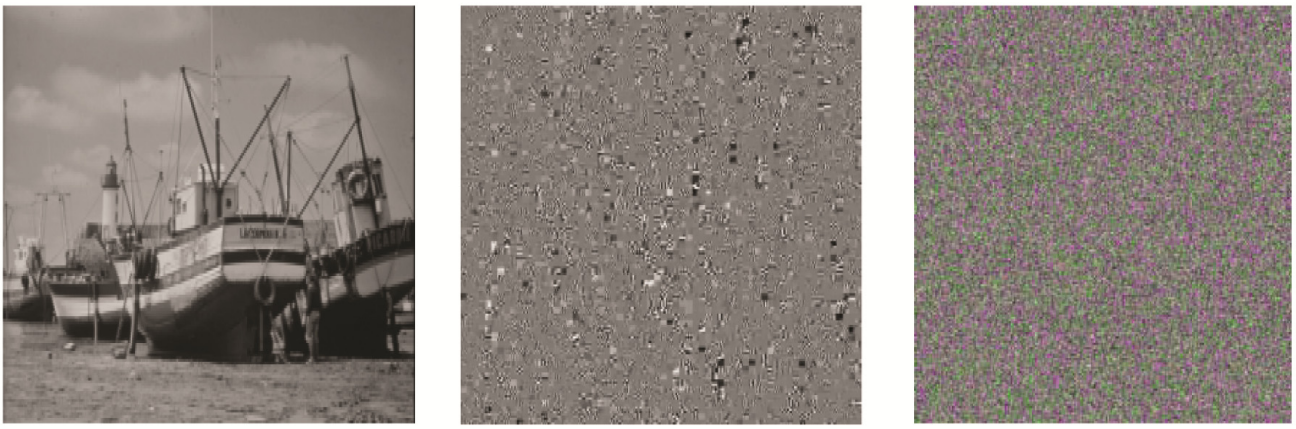


Figure 5. The barbara image is encrypted-compressed and reconstructed (a) the encrypted image (Q = 5), (b) the encrypted image (Q = 25), (c) the encrypted image (Q = 75), (d) the reconstructed image (Q = 5), (e) the reconstructed image (Q = 25), (f) the reconstructed image (Q = 75)

On the other hand, as we know that a high level of security means that the encrypted image does not contain clear parts that can make the attack easier. We use the same image tested in the modified zigzag scan [9] approach. After encryption the encrypted image of our schema is shown in figure 6. The unencrypted parties seen in figure 6. (b) does not appear in figure 6.(c) so we cannot get any information of plain image.



(a) (b) (c)

Figure 6. Comparison between encryption with our approach and modified zigzag scan coding. (a) Original, (b) encryption with modified zigzag scan coding, (c) encryption with our approach

4.2 Compression Performance

To verify the performance of our approach, 7 standard test images (i.e. baboon, Goldhill, Barbara, boat, lena, peppers, areal) of 512 x 512 pixels are considered. The compression ratio CR is computed by equation (5), at image quality Q = 25 and 75 respectively. The results are listed in table 3.

$$CR = \frac{Size\ of\ plain\ image}{Size\ of\ ciphered\ image} \tag{5}$$

Test image	Image quality (db)	CR – our approach	CR – compression and encryption scheme using DCT and SHA-1 [15]	CR – modified zigzag scan coding shema [9]
Barbara	25	3.18	2.07	7.60
Ariel	25	3.17	1.74	5.71
Lena	25	3.21	-	10.32
Peppers	25	3.20	-	8.27
Boats	25	3.11	1.84	8.12
Baboon	25	3.14	1.53	4.97
Goldhill	25	3.20	1.97	8.10
Barbara	75	3.02	-	-
Ariel	75	2.98	-	-
Lena	75	3.10	-	-
Peppers	75	3.09	-	-
Boats	75	3.02	-	-
Baboon	75	2.80	-	-
goldhill	75	3.08	-	-

Table 3. CR of our approach and others approaches

As our approach correlates compression and encryption, and we specifically focused on getting good quality of encryption, we can see that our approach suffers from a degradation of compression performance In comparison with results obtained in the approach proposed in [9], on the other hand, reconstruction quality of our results is clearly better than that of the approach's Yuen et al. [15]. There is always a trade-off between degree of uncertainty for security purpose and length of cipher image: highest security levels imply less compression ratio, while higher compression rates can be achieved if security criterions are reasonable.

4.3 Speed Performance

In order to evaluate the computational complexity, we simulate the processing time of encryption using the proposed method. Our simulate results are shown in table 4.

For comparisons purposes, we also list the processing time of the modified zigzag scan coding [9]. As observed, the processing time of the proposed method is shorter than the modified zigzag scan coding approach. The results shows that the encryption speed differences between the two approaches vary from 0.7 and 0.9 second and this is due to the fact that the modified zigzag scan coding method use multiple modular operations. Conversely the proposed scheme has the advantage of having low computational complexity. The proposed scheme provides faster encryption/decryption operations due to the used model of cellular automata that offer inherent parallelism and simple binary operations.

	Proposed Scheme (Sec)	**Modified Zigzag Scan Coding Scheme (Sec)**
Ariel	0.374	1.361
Baboon	0.265	1.373
Barbara	0.390	1.314
Boat	0.624	1.313
Goldhill	0.405	1.298
Lena	0.371	1.297
Peppers	0.374	1.312

Table 4. Running time of our scheme and modified zigzag scan coding scheme

4.4 Key Sensitivity Analysis

To evaluate sensitivity key, we compare difference between two encrypted images obtained using two keys with one bit change. We use (NPCR) Number of Pixel change to test difference between two images. NPCR is given in (6).

$$NPCR = \frac{\sum_{i=1}^{L}\sum_{j=1}^{H} F(i,j)}{H \times L} \times 100 \tag{6}$$

$$F(j,i) \text{ is defined as } F(i,j) = \begin{cases} 0 \; if \; C1(i,j) = C2(i,j) \\ 1 \; if \; C1(i,j) \neq C2(i,j) \end{cases}$$

Where L and H are the size of image, $C1(i,j)$, and $C2(i,j)$ are pixel values of encrypted images at ith row jth column. Figure 7 shows values of NPCR of encrypted images using our proposed method. An expected NPCR for a good encryption technique is 99,6094% [34] and as observed in figure 7, our proposed method obtains NPCR at least 99,6201%. Therefore, our proposed encryption and compression method is hard to be analysed using similar keys.

4.5 Correlation Analysis

The correlation distribution of two horizontal adjacent pixels, two vertical adjacent pixels and two diagonal adjacent pixels from the original image and encrypted image using our proposed method are shown in Figure 8, where x axis and y axis represent the pixel values chosen randomly and its adjacent pixel values, respectively. We used the pepper image to test correlation between

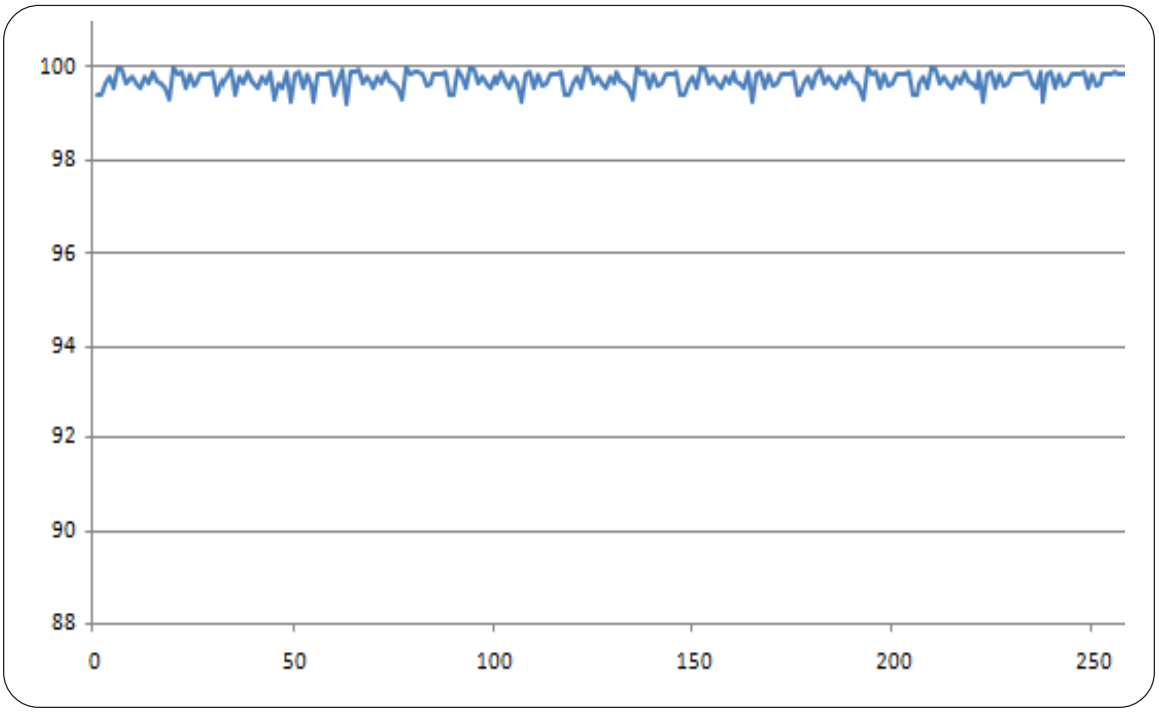
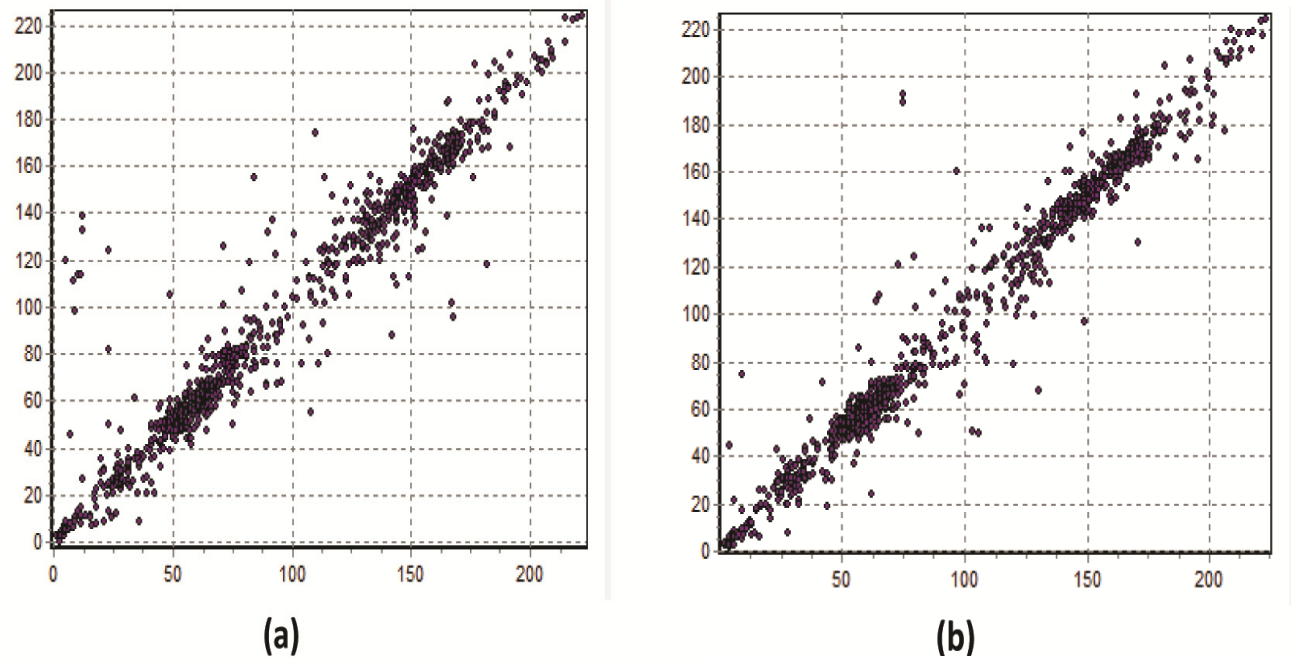


Figure 7. NPCR between images obtained using two keys with one bit change

pixels in three directions.

The correlation distribution of the original image has the high correlation as shown in Figure 8(a-c). From Figure 8(dl), it is clear that patterns of the correlation distribution have non linear form, which means that the encrypted images have the low correlation between two adjacent pixels. Therefore, eavesdroppers could not be able to obtain any useful information from the data with low correlation.



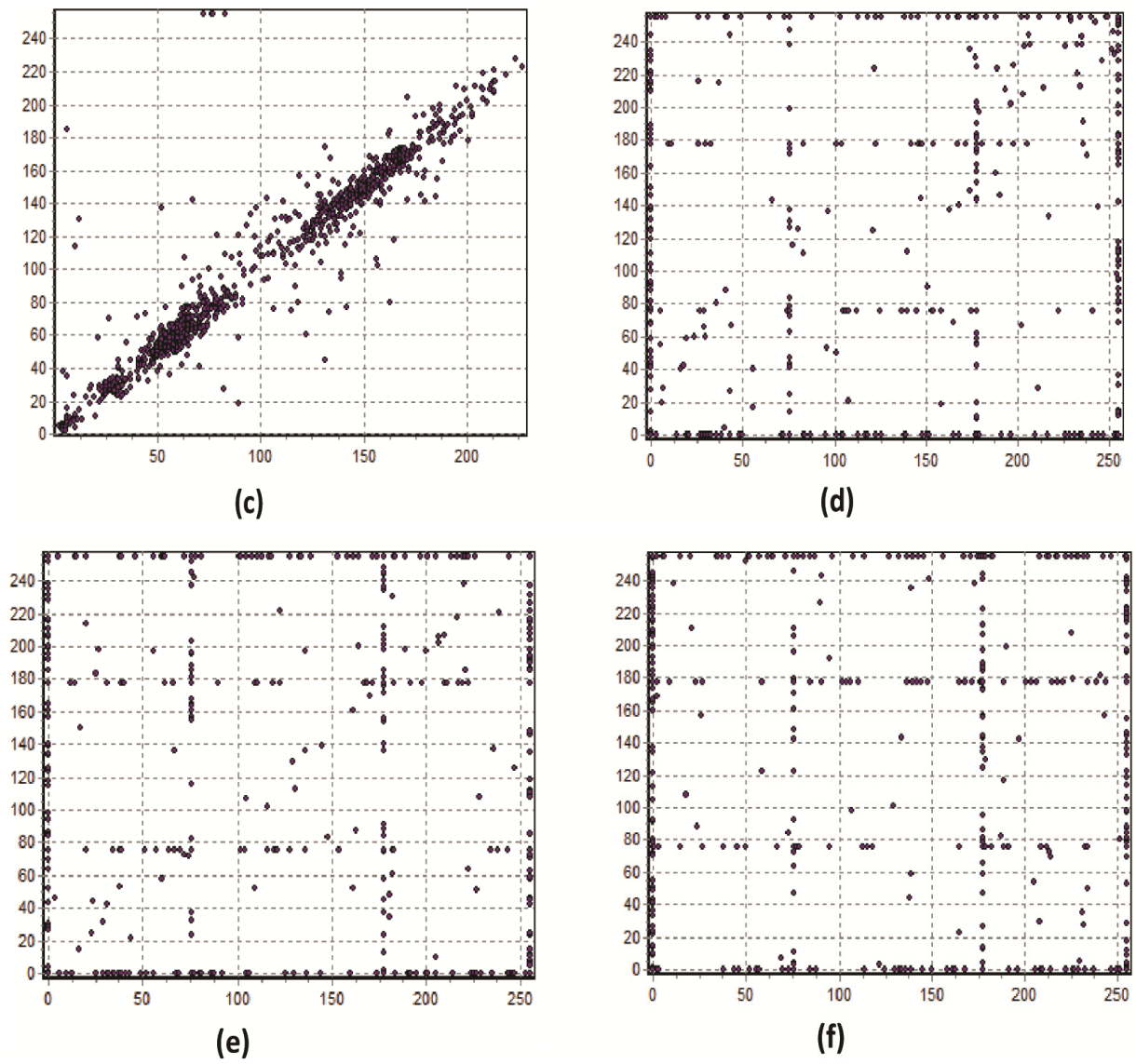


Figure 8. Correlation distribution of tow adjacent pixels (a) the original image (horizontal direction) (b) the original image (vertical direction) (c) the original image (diagonal direction) (d) the encrypted image (horizontal direction) (e) the encrypted image (vertical direction) (f) the encrypted image (diagonal direction)

5. Conclusion

In this paper, we propose an encryption and compression scheme based on chaotic exploration of onedimensional cellular automata, where CA is adopted to scramble sensitive information while keeping the balance between compression performance and security. Experimental results and analyses show the effectiveness and robustness of our proposed scheme.

References

[1] Kingston, A., Autrusseau, F. (2008). Lossless image compression via predictive coding of discrete Radon projections, Signal Processing: *Image Communication* 23, 313–324.

[2] Alkholidi, A., Alfalou, A., Hamam, H. (2006). A new approach for optical colored image compression using the JPEG standards. *Signal Process* 87, 569–583.

[3] Alfalou, A., Alkholidi, A. (2005). Implementation of an all-optical image compression architecture based on Fourier transform which will be the core principle in the realization of DCT. Proc SPIE 5823, 183–190. (jepeux le remplacé par un de ces bibliog).

[4] Cagnazzo, M., Cicala, L., Poggi, G., Verdoliva, L. (2006). Low-complexity compression of multispectral images based on classified transform coding. *Signal Process Image Commun* 21, 850–861.

[5] Ayoub, F., Singh, K. (1984). Cryptographic techniques and network security, *IEE Proceedings* (7) 131, 684–694.

[6] Izmerly, O., Mor, T. (2006). Chosen ciphertext attacks on lattice-based public key encryption and (non-quantum) cryptography in a quantum environment. *Theoretical Computer Science,* 367, 308–323.

[7] Wong, K. W., Yuen, C. H. (2008). Embedding compression in chaos-based cryptography, IEEE Transactions on Circuits and Systems II: *Express Briefs* (11) 55, 1193–1197.

[8] Ong s. and al Beyond format-compliant encryption for JPEG image, Signal Processing: *Image Communication* 31(2015) 47–60

[9] X.-y. Ji and al. A new security solution to JPEG using hyper-chaotic system and modified zigzag scan coding, *Commun Nonlinear Sci Numer Simulat* 22 (2015) 321–333

[10] Bahrami, S., Naderi, M. (2013). Encryption of multimedia content in partial encryption scheme of DCT transform coefficients using a lightweight stream algorithm, *Optik* 124, 3693– 3700.

[11] Hermassi, H. (2010). Joint compression and encryption using chaotically mutated Huffman trees, *Commun Nonlinear Sci Numer Simulat* 15, 2987–2999.

[12] Li, W., Yuan, Y. (2007). A leak and its remedy in JPEG image Encryption, *International Journal of Computer Mathematics*, (9) 84, 1367–1378.

[13] Kailasanat.han, C., Safavi Naini, R. (2002). Compression Performance of JPEG Encryption Scheme, 2 , 1329-1332

[14] Bhargava, B. K., Shi, C., Wang, S.-Y. (2004). MPEG video encryption algorithms, Multimed. Tools Appl. (1) 57–79.

[15] Yuen, C.-H., Wong, K.-W. (2011). A chaos-based joint image compression and encryption scheme using DCT and SHA-1, *Applied Soft Computing* 11, 5092–5098.

[16] Zhou, J. (2007). Security Analysis of Multimedia Encryption Schemes Based on Multiple Huffman Table, *IEEE Signal Processing Letters,* (3) 14, 201-204.

[17] Takayama, M., Tanaka, K., Takagi, K., Nakajima, Y. (2008). A scalable video scrambling method in mpeg compressed domain. *International Symposium on Communications*, (2008). 1035–1040.

[18] Zhang, X., Wang, C., Zhong, S., Yao, Q. (2013). Image encryption scheme based on balanced two dimensional cellular automata. *Mathematical Problems in Engineering*, 2013, 10 p.

[19] Chen, T. (2016). Image encryption and compression based on kronecker compressed sensing and elementary cellular automata scrambling, *Optics & Laser Technology*, 84, 118–133.

[20] Tralic, D., Grgic, S., Robust. (2016). Image Encryption Based on Balanced Cellular Automaton and Pixel Separation. *Radio Engineering*, (3) 25, 548-555.

[21] Hortensius, P. D., McLeod, R. D., Card, H. C. (1989). Parallel random number generation for VLSI systems using cellular automata, *IEEE Trans. on Computers* 38. 1466-1473.

[22] Nandi, S., Kar, B. K., Chaudhuri, P. P. (1994). Theory and Applications of Cellular Automata in Cryptography, *IEEE Trans. On Computers*, v. 43, (December) 1346-1357.

[23] Tomassini, M., Perrenoud, M. (2000). Stream Ciphers with One- and Two-Dimensional Cellular Automata, in M. Schoenauer at al. (Eds.) *Parallel Problem Solving from Nature - PPSN VI*, LNCS 1917, Springer, 2000, *p*. 722-731.

[24] Howard Gutowitz, *A Massively Parallel Cryptosystem Based on Cellular Automata*,(1994)

[25] Kang, B.-H. A Pseudorandom Number Generator Based on Two-Dimensional Programmable Cellular Automata, Doctor's Thesis, Kyungpook National University (June 2007)

[26] Quieta, M. T. R., Guan, S.-U. (2005). Optimization of 2D Lattice Cellular Automata for Pseudorandom Number Generation. *International Journal of Modern Physics C*, 16 (3) 479–500.

[27] Bouvry, P., Seredynski, F., Zomaya, A. Y. (2003). Application of cellular automata for cryptography. *In:* PPAM, p.447-454.

[28] Seredynski, M., Bouvry, P. (2004). Block cipher based on reversible cellular automata. Evolutionary Computation, 2004. CEC2004. Congress on, 2. 2138-2143, 2, 19- 23. (June).

[29] Srebrny, M., Such, P. (2003). Encryption using two-dimensional cellular automata with applications. *Artificial intelligence and Security in Computing Systems*, p. 203-215.

[30] Tao, R., Chen, S. (1999). On finite automaton public-key cryptosystem. Theoretical Computer Science, 226(1- 2) 143-172.

[31] Guan, P. (1987). Cellular automata public key cryptosystem. Complex Systems, 1. 51-57.

[32] Kari, J. (1992). Cryptosystems based on reversible cellular automata. Manuscript.

[33] Zeng, Z., Wang, N. (2003). A new method based on chaotic sequence of image encryption. Proceedings of SPIE, the Third International Conference on Photonics and Imaging in Biology and Medicine. (2003) 285- 289

[34] Wu, Y., Noonan, J. P., Agaian, S. (2011). NPCR and UACI randomness tests for image encryption, Cyber Journals: *Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications* (*JSAT*). (2011)31–38.