Research Article

An Image Cryptosystem based on Elementary Cellular Automata with Integrity Checking

Doi: 10.30508/kdip.2024.448515.1097

Mona Kordestani[1]

Abstract

This paper presents a novel image cryptosystem using a specific feature of boundary elementary cellular automata with permutation-diffusion architecture. The characteristics of some periodic boundary elementary cellular automata (CAs) with the length of 8 have been studied over the past years. The State-transition diagram shows that some elementary cellular automata (ECA) rule lead to some state attractors that are suitable to use to implement an encrypting function in order to transform the pixels' values while meeting the basic requirements of the encryption scheme. Minimizing computational overhead, the production of these attractors could be done by using an initial global state of the CA solely, and the implementation of which needs no extra hardware cost. Correspondingly, the proposed new image cryptosystem includes an encryption method based on both permutations of the image pixels and the replacement of the pixel values. The permutation method is optional and can be done by every permutation algorithm. Simulation results revealed this fact that the proposed CA-based image cryptosystem shows strong performance in terms of encryption and decryption. The outstanding characteristics of the proposed image encryption method are as follows: lossless and symmetric private key encryption, low data expansion, the possibility of encryption when there is more than one image with the use of just one key image, large keyspace, and checking data integrity.

Keywords: Attractors, Cellular automata, Data integrity, Image encryption

1- Department of Computer Engineering, Islamic Azad University, Mashhad Branch, Mashhad, Iran

1- Introduction

Never before has so much secret data been sent to others during human interactions. Thus, Confidentiality of data and information is a real necessity nowadays. Cryptography is one of the methods that could be used as a protection tool. Visual data has some specific features, such as the huge amount of data and the high correlation among pixels. Regarding these features, the common and traditional cryptosystems are not suitable for encrypting secret images, so specific methods must be adopted. We propose a new image encryption/decryption method based on specific features of CA like state attractors in this paper. The permutation of the image pixels is done by using a simple permutation method and the replacement of the pixel values is done by using a state attractors feature. This method has the ability to check the data integrity of the encrypted images. Our proposed method has a low data expansion without any distortion between the decrypted image and original image and the ability of a decryption arbitrary number of images by one secret key. This method provides a large keyspace which is redounded high security.

The rest of the paper is organized as follows: Section 2 describes the behavior of state attractors and their properties. The proposed method is described in Section 3. Security issues regarding the application of the proposed method are checked in Section 4. Section 5 gives statistical analysis, simulation results and analysis, and performance comparison of the proposed method with other published algorithms. An implementation of the proposed method on specific hardware is discussed in Section 6. Finally, conclusions are provided in Section 7.

2- literature Review

In the last few years, the rapid spread of the internet and using visual multimedia data over this unsecured communication channel, has created a demand calling for an image encryption method that is secure and effective. In the past, however, encryption methods were used by military applications, nowadays it is becoming necessary in more and more ordinary fields of life. With regard to the features of images, such as bulk data capacity and the high correlation among pixels, traditional cryptosystems are not appropriate for practical image encryption.

Traditional video watermarking algorithms are mostly based on discrete-transform domains, such as DCT and DWT. In paper (Li, Kim, & Wang, 2017), proposed a video watermarking method by employing three-dimensional (3D) cellular automata (CA) transform and perfect-reconstruction integral imaging (PRII). This paper (Hanis, & Amutha, 2018) proposed a key generation algorithm and a double image encryption scheme with combined compression and encryption. Also, cellular automata-based diffusion is performed on the resultant image to strengthen the security further. In paper (Ahmad, Doja, & Beg, 2021) aims to investigate and analyze the security of this image cryptosystem in terms of its defects and attack resistivity. The security analysis uncovers that cryptosystem holds certain serious security defects and is incapable to secure encrypted content. Some of the studies proposed an attack model to break the security of an image cipher entitled "Reusing the permutation matrix dynamically for efficient image cryptographic algorithm" developed by Chen et al. This paper (Diab, & El-semary, 2018) also suggested an image cryptosystem to overcome the security inadequacies of the scheme under Study.

Over the past years, lots of image encryption methods have been invented. The most common image encryption methods are divided into 4 categories: SCAN-based methods (Alexopoulos, Bourbakis, & Ioannou,1995., Jawad, 2021), chaos-based methods (Das, & Adhikari, 2010., Vaseghi, Mobayen, Hashemi, & Fekih, 2021), tree structure-based methods (Li, Knipe, & Cheng,1997., Zeng, & Wang, 2021) and other systematic methods (Bakhshandeh, & Eslami, 2013., Wu, & Moo, 1999). Each has its strengths and limitations in terms of security, speed, and resulting stream size metrics. A new Multi-Stage Multi-Secret Image Sharing (MSMSIS) scheme, based on polynomial sharing and cellular automata, is proposed in this paper. Also, a particular type of cellular automata is used to guarantee that the secrets are only recovered according to a pre-specified order (Zarepour-Ahmadabadi, Shiri-Ahmadabadi, & Latif, 2018). This study (Ferretti, Marchetti, Andreolini, & Colajanni, 2018), proposed an original solution based on encrypted Bloom filters that address the latter problem by allowing a cloud service used to detect unauthorized modifications to his outsourced data.

Considering the huge amount of image data, the image cryptosystems should be fast enough (Li, Peng, Tan, & Li, 2020., Talhaoui, & Wang, 2021). CAs due to easy hardware implementation are appropriate candidates for image cryptosystems (Lafe, 1997). Key-based cryptography algorithms can be divided into two classes, private and public keys (Pieprzyk, Seberry, & Hardjono, 2004). Cellular Automata (CAs) were proposed for public-key cryptosystems by Guam (Guam, 1987) and Gutowitz (Gutowitz, 1993). In this document, the proposed image encryption method is focused on a symmetric key cryptosystem. In the context of a symmetric key cryptosystem, CAs have been studied by wolfram (Wolfram, 1986) at first, and later by Hortensius (Hortensius, McLeod, & Card, 1989), Nadi (Nandi, Kar, & Chaudhuri, 1994), Sipper (Sipper, & Tomassini, 2000), and Seredynski (Seredynski, Bouvry, & Zomaya, 2004). Regarding image encryption, some works based on CA already have been done. In (Seredynski, Bouvry, & Zomaya, 2004), in a Vernam cipher cryptography method, for producing the bitstream of the CAs were used. In (Yu, Lu, Leung, & Chen, 2005) a similar group of basic functions which are generated from the CAs' states evolvement, is used to encrypt data. In (Yu, etal 2005), an image encryption method based on a permutation of the pixels of the image and replacement of the pixel values has been proposed. For permutation, scan pattern algorithms were used. However, each of them has its own strengths and weaknesses in terms of security level, speed, resulting in stream size metrics, Needing high computational and complicated process for permutation and CA's setup and evolvement, Using different secret keys for each or a finite number of images which should be encrypted, High data expansion, Difference and distortion between decrypted image and secret image and Impossibility of checking data integrity at the receiver side. In (Enayatifar, Sadaei, Abdullah, Lee, & Isnin, 2015) a novel method based on a hybrid model of the Tinkerbell chaotic map, deoxyribonucleic acid (DNA), and CA is proposed. In (Del Rey, Sánchez, & De La Villa Cuenca, 2015) an image cryptosystem for RGB digital images is proposed. The proposed algorithm includes two iterative sections: the confusion phase ruled by the 2D chaotic Cat map, and the diffusion phase governed by reversible memory CA. In (Li, Sun, Li, & Chen, 2017) an image encryption algorithm based on non-uniform second-order reversible cellular automata (CA) which consist of two-state cells and use the Moore neighborhood is proposed which uses the balanced CA rules that have higher randomness than unbalanced rules. Moreover, in (Chai, Gan, Yang, Chen, & Liu, 2017) by employing the memristive hyperchaotic system, cellular automata (CA), and DNA sequence operations, an image cryptosystem is presented, which consists of the diffusion process. SHA 256 hash function is used to give the secret key and compute the initial values of the chaotic system.

State Attractor

The behavior of a number of elementary cellular automata (ECA) rules has been investigated over the past years due to their specific characteristics. Since their implementation needs a simple hardware structure and requires minimized computational resources, it is chosen to investigate ECA rules.

When dealing with finite CA, cyclic boundary conditions are typically applied. A CA with cyclic boundary in CA means that the left neighbor of the left-most cell is the right-most cell, and the right neighbor of the right-most cell is the left-most cell. In this proposed method, cyclic boundary conditions are chosen. Constructing their state transition diagrams yields information on the global states that result when the automaton starts its evolution from a given preliminary configuration. It is clear that for an ECA with the length of 8, there are 256 different global states. The aim of this research was finding ECA rules that fulfil the following criteria: consecutive states on the state transition diagram must be produced; the states are organized in circles such that after the application of all states in the set; and the ECA should be prepared to repeat the same procedure.

In the state transition diagrams, it must be noticed that some rules such as rule 42 definitely fulfil the criteria: the states produced by the evolution of the ECA under these rules are prepared in circles. We call these circles state attractors. A close examination of rule 42 is performed. Fig. 1 shows the state transition diagram for rule 42, where the states are shown in binary-coded decimal form for easy representation; and each attractor containing 2, 4, or 8 states (except for attractor 0). The states are not shown in the figure to enter one of these attractors through some or the other states on these attractors after a number of transitions. Among these state attractors, the attractors containing 8 states are capable of recurrence of a data with the integer value between 0 and 255 after the data consecutively circling through 8 states on the attractor, see formula:

$$d \oplus state(1) \oplus \dots \oplus state(8) = d$$

Where $\oplus$ is bitwise XOR, for example; pix1=210 andpix2=169 are two pixels of a grayscale image, the bit wise XOR operation of the two pixels as follows:

$pix1 \oplus pix2 = (210)_{10} \oplus (169)_{10} = (11010010)_2 \oplus (10101001)_2 = (01111011)_2 = (123)_{10}; \ state(i), 1 \le i \le 8$, are the 8 consecutive states of an attractor, d is an integer with a value between 0 and 255. The XOR operation can start with any state, i.e. the initial state, state (1) can be any state (i) $1 \le i \le 8$ on the attractor. This event meets an elementary requirement of the encryption method. Depending only on the ECA rule and the given initial state (initial configuration), the generated attractor could be used as a keystream as in the field of stream ciphers, to encrypt and decrypt data. Let $t \ (1 \le t \le 8)$ be the encryption time. Begin with any state, say state (i), on an attractor, applying an XOR operation to d with the consecutive t states, the encrypted data en_d is obtained; keeping on applying an XOR operation to en_den_d with the residuary consecutive 8-t states on the attractor, will return back to dd. The encryption/ decryption process can be seen in the formula:

$$d \oplus state(i) \oplus \dots \oplus state(t) = en\_d$$

$$en\_d \oplus state(t+1) \oplus \dots \oplus state(i-1) = d$$

After analysis of their state transition diagrams, it is also realized that other 8 length ECA rules show similar behavior, such as rules 56,112, 120, 175, 248, and so on.

The Proposed Scheme

In this section, a novel image cryptosystem scheme is proposed based on one-dimensional elementary state attractors. The basic idea of the proposed image cryptosystem method is to rearrange the pixels of the image and change the pixel values. The pixel rearrangement is performed by the use of the permutation matrix. The pixel values are changed by using the specific characteristic of state attractors and applying an XOR operation after doing a bitwise XOR action with values of state attractor. The scheme can be used to encrypt an image with the size $M \times N$. This is a color image and is comprised of 3 layers; Red, Green, and Blue layer. The proposed method gives a number of advantages. First of all, transmission security can be guaranteed by the large complexity required in order to have access to the data. Moreover, in terms of complexity, the proposed method is excellent. The results can be achieved by the traditional One-Time Pad (OTP) method overcoming the constraint of different keys applied for each image. The proposed method can encrypt an arbitrary number of images with the use of just one key.

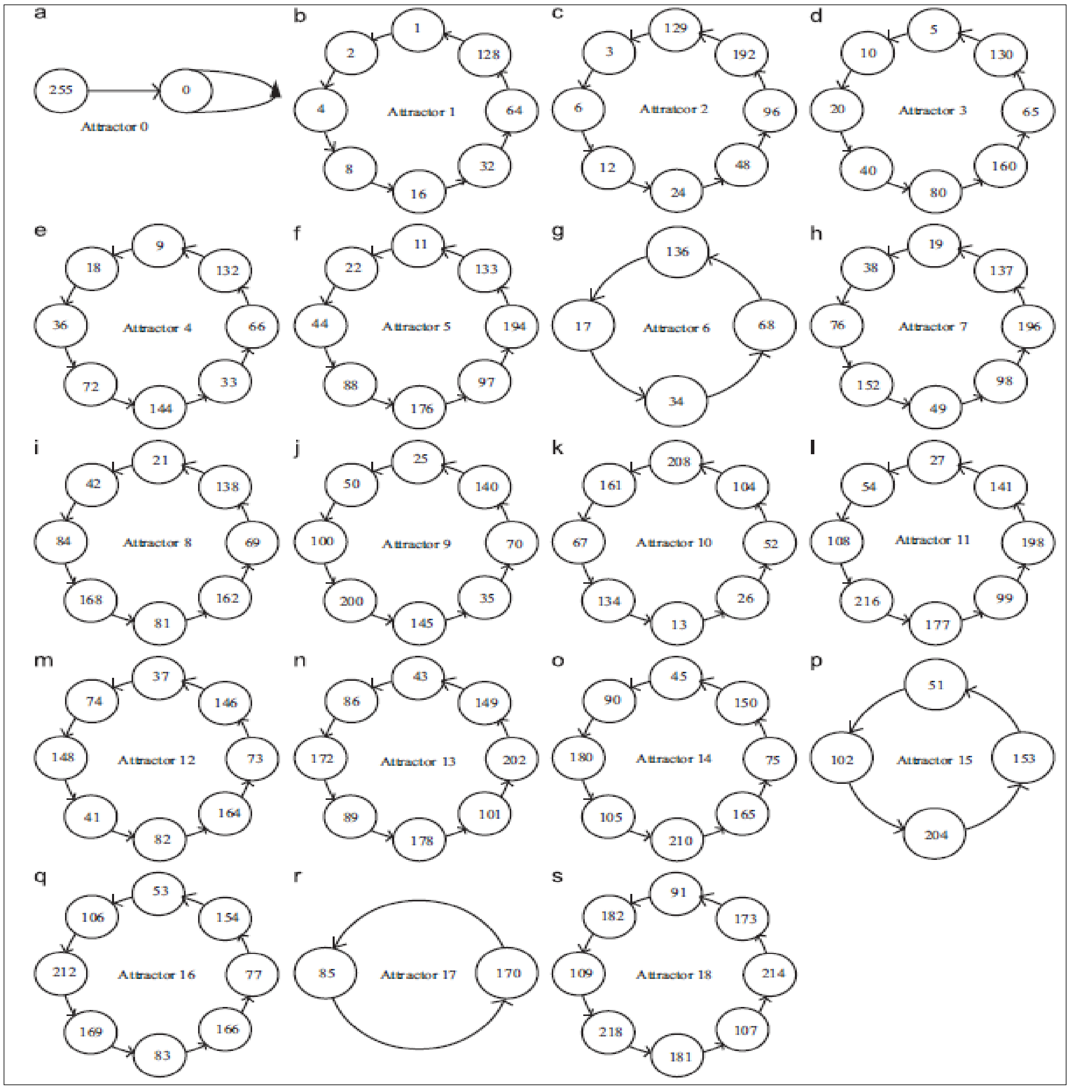


Fig. 1. State transition graph for rule 42

Moreover, an important feature that none of the traditional image encryption methods can exhibit is the ability to check the integrity of different parts of the decrypted image with arbitrary accuracy. Furthermore, an important feature of the proposed method is its attribute to deteriorate elements of the encrypted image that would possibly reveal information relative to the nature of the image that was encrypted. Our scheme consists of 3 phases: (1) the encryption phase in which the secret image is encrypted by using permutation matrixes and one-dimensional Linear Memory Cellular Automata (LMCA), (2) the decryption phase in which the encrypted image is decrypted to the original secret image, (3) the integrity test phase in which the receiver can accept the integrity of the image.

Notations

SI	The secret image	$B_i^{'}$	The ith block of permuted secret image pixels
M	Number of rows in secret image	H_i	The ith hash value corresponding to Bi
N	Number of columns in secret image	H'_i	The ith hash value corresponding to Bi

PM_I	The permutation matrix for permuting the pixels of secret image	R	The rule of elementary CA
PM_H	The permutation matrix for permuting the hashes value	A_{ID}	The number of states used for bitwise XOR action
N_B	Number of blocks	B_{en}	The encrypted block of bits
SK	The secret key	B'_{en}	The permuted encrypted block of bits
B_i	The ithith block of secret image pixels	State(i)	The ith value in state attractor circle

Image Encryption

The proposed method is a symmetric private key security method. It means that the same key is required for encryption/decryption, and of course both sender and receiver must know the key. For encryption, the proposed method uses 5 keys which are generated at the sender site. The first key is a permutation matrix PM_I which is used to rearrange the pixels of the secret image, the second key is a permutation matrix PM_H which is used to permute the hashes value corresponding to the blocks of pixels, the third key is a seed of PseudoRandom Number Generator (PRNG) which can generate high-quality random numbers between 0 and 255, the R is the rule of elementary CA used to produce state attractor and A_{ID} is the number of states used for bitwise XOR action in the encryption process. These keys are sent to the receiver side via a secure channel. In this phase at the sender site, the user performs the following steps to encrypt the secret image. The encryption procedure is illustrated in Fig. 2. Since the secret image SI is a color image and consists of 3 layers, the encryption procedure must be applied to all layers. The following steps are done for each layer individually, and finally, these layers merged together to construct the encrypted image. The user at sender site:

1-Divides the secret image SI into N_B blocks of 7 bits. N_B, the number of secret image blocks is defined as follows:

$$N_B = \left|\frac{M \times N}{6}\right|$$

2-For each blocks Bi where $1 \leq i \leq N_B$ computes the hash of 7 bits belonging to this block as follows:

$$H_i = B_i(1) \oplus \dots \oplus B_i(6)$$

Where $B_i(k)$ is the kth bit value of i^{th} block of image bits.

3-Permutes the original secret image according to the PM_I permutation matrix and after that divides the permuted secret image into N_B blocks of 7 pixels to obtain $B'_1, B'_2, \dots, B'_{NB}$

4-Permutes the hashes value $H_1, H_2, \dots, H_{N_B}$ obtained from step (2) by a permutation matrix PM_H PM_H. After this step user at the sender site has $H'_1, H'_2, \dots, H'_{N_B}$.

5-Initializes the initial configuration of one-dimensional elementary CA with the value of rule R and by the evolution of CA according to rule R, obtain the states of the state attractor as follow: $state(1), \dots, state(k)$ where $1 \leq k \leq 8$.

6-Now the bits of each block B'_i, and a bit b are calculated by hashing all bits of $B'_{i-1}B'_{i-1}$ put together to make a value $Block_{value}$. Notice that if i is equal to 1, the value of b is random and public.

7-At the sender site, the user generates the secret key set by using PRNG with seed. Then the encrypted value of the 6-bits block of SI is calculated as follows:

$$B_{en} = Block_{value} \oplus Sk(i) \oplus State(1) \oplus \dots \oplus state(A_{ID})$$

8-Steps (5) to (7) are repeated for all permuted secret image blocks.

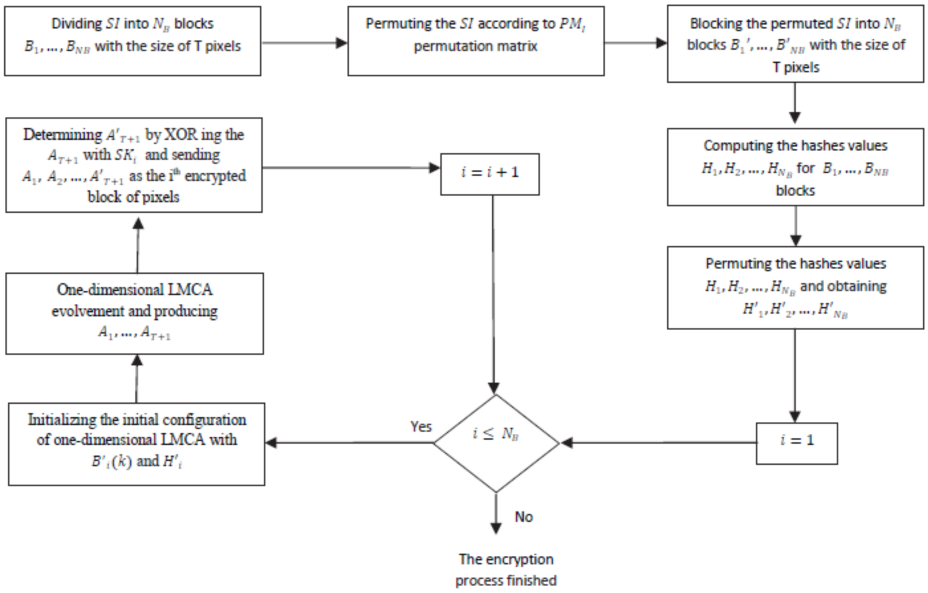

Dividing SI into N_B blocks $B_1, \dots, B_{NB}$ with the size of T pixels
Permuting the SI according to PM_I permutation matrix
Blocking the permuted SI into N_B blocks $B_1', \dots, B'_{NB}$ with the size of T pixels
Computing the hashes values $H_1, H_2, \dots, H_{N_B}$ for $B_1, \dots, B_{NB}$ blocks
Permuting the hashes values $H_1, H_2, \dots, H_{N_B}$ and obtaining $H'_1, H'_2, \dots, H'_{NB}$
$i = 1$
$i \leq N_B$
Yes
No
The encryption process finished
Initializing the initial configuration of one-dimensional LMCA with $B'_i(k)$ and H'_i
One-dimensional LMCA evolvement and producing $A_1, \dots, A_{T+1}$
Determining A'_{T+1} by XOR ing the A_{T+1} with SK_i and sending $A_1, A_2, \dots, A'_{T+1}$ as the ith encrypted block of pixels
$i = i + 1$

Fig. 2. The flowchart of image encryption procedure for each image layer.

Image Decryption

The decryption process is illustrated in Fig. 3. The following methodology is followed at the receiver site by the user:

1-Receives the keys via a secure channel and generates the secret key set

2-SK by using PRNG with seed.

3-Receives each encrypted block value:$B_{en}(1), \dots, B_{en}(N_B)$ and determines the A_{N_B} corresponding to each$N_B$values as follows:$A_i = SK(i) \oplus B_{en}(i)$, where $1\square\leq i\square\leq N_B 1\square\leq i\square\leq N_B$.

4-Initializes the initial configuration of one-dimensional elementary CA with the value of rule RR and by the evolution of CA according to rule RR, obtain the states of the state attractor as follows:

5-$state(1), \dots, state(k)$ where $1 \leq k \leq 8$.

6-Calculate the decrypted 6-bit block as follows:

$$B'_{de}(i) = (A_i \oplus Sk(i) \oplus State(A_{ID}+1) \oplus \dots \oplus state(length\ of\ state\ attractor))$$

7-

$$H'_i = (A_i \oplus Sk(i) \oplus State(A_{ID}+1) \oplus \dots \oplus state(length\ of\ state\ attractor))$$

8-And

$$H'_i = (A_i \oplus Sk(i) \oplus State(A_{ID}+1) \oplus \dots \oplus state(length\ of\ state\ attractor))$$

9- The blocks of 6 values,$B^{'}de(1),\ B^{'}de(2), \dots, B^{'}de(N_B) B^{'}de(1),\ B^{'}de(2), \dots, B^{'}de(N_B)$ are put together to construct the permuted secret image.

10-Rearranges the bits of the permuted secret image according to the permutation matrix PM_I and after the original secret image is determined.

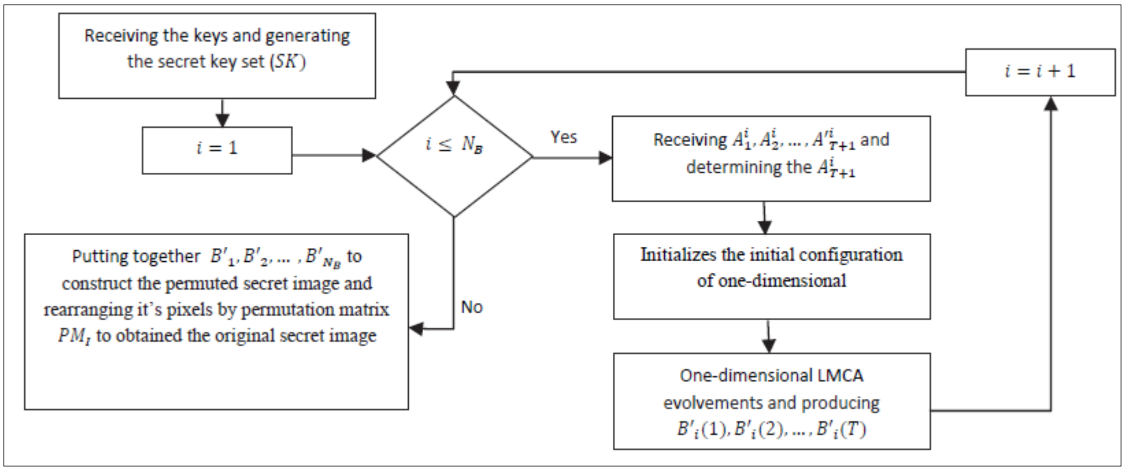

Receiving the keys and generating the secret key set (SK)
$i = 1$
$i \leq N_B$
Yes
No
$i = i + 1$
Receiving $A_1^i, A_2^i, \ldots, A'^i_{T+1}$ and determining the A_{T+1}^i
Initializes the initial configuration of one-dimensional
One-dimensional LMCA evolvements and producing $B'_i(1), B'_i(2), \ldots, B'_i(T)$
Putting together $B'_1, B'_2, \ldots, B'_{N_B}$ to construct the permuted secret image and rearranging it's pixels by permutation matrix PM_I to obtained the original secret image

Fig. 3. The flowchart of image decryption procedure for each image layer.

Integrity Checking Process

The integrity test procedure is illustrated in Fig. 4. For checking the integrity of the received image, the user at the sender site performs the following steps:

1-Rearranges the values;$H'1, H'2, \ldots, H'NBH'1, H'2, \ldots, H'NB$ according to the permutationmatrix $PM_H PM_H$ to obtain hash values $H_1, H_2, \ldots, H_N H_1, H_2, \ldots, H_N$.

2-Divides the image produced from the image decryption process into $N_B N_B$ blocks with the size of 6 bits

3-Determines the hash value for each block by using an XOR operation as follows:

$h_i = B_i(1) \oplus B_i(2) \oplus, \ldots., B_i(T)$

where $B_i(k) B_i(k)$ is the $kth kth$ bit value of $i^{th} i^{th}$ block of the image.

4- Compares the hash value $H_i H_i$ with the new hash value $h_i h_i$. If these two values are not the same, the integrity test for new blocks of pixels $B_i B_i$ is fail.

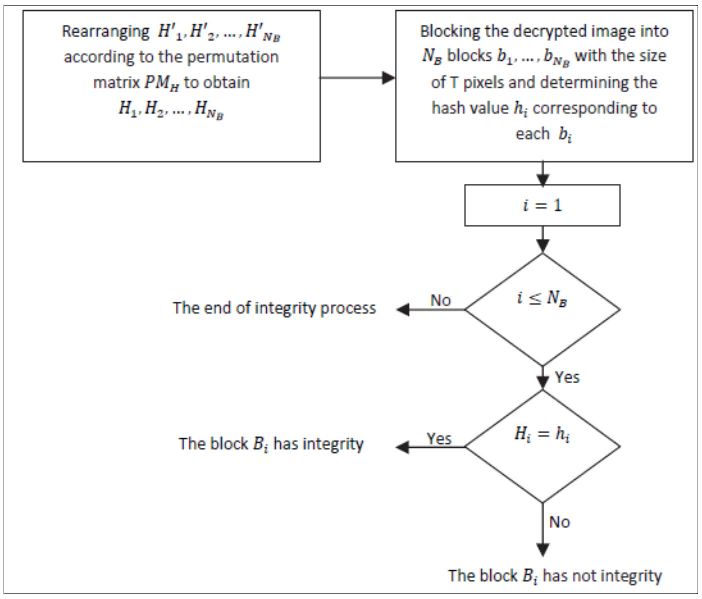

Rearranging $H'_1, H'_2, \ldots, H'_{N_B}$ according to the permutation matrix PM_H to obtain $H_1, H_2, \ldots, H_{N_B}$
Blocking the decrypted image into N_B blocks $b_1, \ldots, b_{N_B}$ with the size of T pixels and determining the hash value h_i corresponding to each b_i
$i = 1$
$i \leq N_B$
No
The end of integrity process
Yes
$H_i = h_i$
Yes
The block B_i has integrity
No
The block B_i has not integrity

Fig. 4. The flowchart of integrity test procedure for each decrypted image layer.

3- Security Analysis

An ideal encryption scheme should be strong and at least robust against all kinds of well-known attacks and should have a large enough keyspace to make brute-force attacks impossible.

To show that the proposed scheme is secured against the most common attacks, some security analyses, such as statistical analysis, sensitivity analysis with respect to the key and plain text, and keyspace analysis have been done on a proposed image encryption scheme.

Exhaustive Search Attack

A Brute force attack is a trial and error method of trying every possible combination of characters with the hope of eventually guessing the password correctly. The security of the proposed image is based on the permutation matrixes PM_H,PM_I,SK, R and A_{ID}. As mentioned above,PM_I is used to rearrange the pixels of an image with size$\times$ N . PM_Hpermutes the hash values belonging to blocks of the image. SK Contains some random values produced by PRNG with seed. R is the rule of elementary CA used to produce the state attractor and A_{ID} is the number of states that should be attended in XOR action during the encryption process. Given such a key, a very large number of resultant potential security keys are available $(N \times M)!(NB)!(256)(8)(256)$. This keyspace is large enough to make brute-force attacks infeasible.

Known Plain Text (DATA image)-Cipher Text (ENCRYPTED image) Attack

The attacker has access to one or more cipher-images and the corresponding plain images. The purpose is to somehow find the secret key. In this security test, the complexity of the keys used in our proposed method appears to be greatly increased. Suppose the attacker possesses an encrypted image produced for a known data image with size $M \times N$. This encrypted image should be divided into blocks with the size of T pixels by the adversary. After that, the adversary could determine the hash values corresponding to each block. But to determine which hash value is belonged to specific blocks of the original secret image, the adversary needs to know the permutation matrixPM_H. For finding the values of SK set the adversary should check values 0, 1, ..., 255 but for finding the appropriate values he/she should know the permutation matrixes PM_I, PM_H, A_{ID}, R. So far using Known plain text (DATA image)-Cipher Text (ENCRYPTED image) attack to find the keys, the adversary should use exhaustive search and tests every possible combination for PM_I, PM_H, A_{ID}, $R$and$SK$.

Differential Attack

The differential attack is a general name of attacks/cryptanalysis applicable primarily to block ciphers working on binary sequences. The discovery of differential attack is typically attributed to Eli Biham and Adi Shamir. The NPCR and UACI are designed to check the number of changing pixels and the number of averages changed intensity between cipher-images, respectively, when the difference between plaintext images is negligible (usually a single pixel). Although these two tests are compactly determined and are easy to compute, test scores are difficult to interpret in the sense of whether the performance is good enough. They are defined as follows (Hortensius et al., 1989):

$$\mathrm{NPCR} = \frac{\sum_{i,j} D(i,j)}{N \times M} \times 100$$

$$\mathrm{UACI} = \frac{1}{N \times M}\left|\sum_{i,j} \frac{|C_1(i,j) - C_2(i,j)|}{255}\right| \times 100$$

where $C1$ and $C2$ are the two cipher-images of which the corresponding plain-images have the only one-pixel difference, the grey-scale values of the pixels at position (i,j) of $C1$ and $C2$ are represented as $C1(i,j)$ and $C2(i,j)$, respectively; M and N are width and height of the cipher-image, respectively; $D(i,j)$ is defined by $C1(i,j)$ and $C2(i,j)$, namely, if $C1(i,j) = C2(i,j)$,

$D(i,j) = 1$; Otherwise, $D(i,j) = 0$. First, plain-image is encrypted. Then, a pixel in that image is randomly chosen and toggled. The revised image is encrypted again by using the same key to generate a new cipher-image. Finally, the $NPCR$ and $UACI$ values are computed. This kind of test is done several times with different images. The resulting maximum, minimum, and average $NPCR$ and $UACI$ values are tabulated in Table 1. The simulation results show that the proposed cryptosystem resists the differential attack.

Table 1. The maximum, minimum and average NPCR and UACI.

	max(%)	min(%)	average(%)
NPCR	99.4004	99.316	98.3582
UACI	33.9512	33.6461	33.7986

Difference between the Encrypted Image and Secret Image

The Peak Signal-to-Noise Ratio (PSNR) can be used to evaluate an encryption method. PSNR reflects the quality of encryption. It is a measurement that demonstrates the changes in pixel values between the plain-image and the cipher-image to distinguish the difference between the ENCRYPTED image and the DATA one we adopt the function of peak signal to noise ratio PSNR (Kamali et al., 2010; Enayatifar et al., 2015). The PSNR computation is derived from the following equation:

$$\mathrm{PSNR} = 20 \times \log_{10}\left(\frac{(2^{n}-1)\sqrt{N \times N}}{\sum_{i=1}^{N}\sum_{j=1}^{N}|\mathrm{secretimage}(i,j) - \mathrm{encryptedimage}(i,j)|^{2}}\right)$$

where N is defined as the square dimension of the images and n is the number of bits representing a pixel. When 8-bit images are used, the greatest distance between images is computed at 0 dB, while when matching the images the value extends infinitely. The results for the images in Fig. 5 are represented in Table 3.

4- Statistical Analysis

It is well known that passing the statistical analysis on ciphertext is of crucial significance for a cryptosystem. In fact, a perfect cipher should be robust against any statistical attack. To evaluate the security of the proposed image encryption scheme, the following statistical tests are done.

Histogram

Encrypt the test image, Lena, and then plot the histograms of the plain-image channels and cipher-image channels as shown in Figs. 6 (a) and (b), respectively. The latter figure indicates that the histogram of the cipher-image is almost at, implying a good statistical property.

Correlation of Adjacent Pixels

The correlation coefficient is a useful measure to evaluate the encryption quality of any cryptosystem (Elashry et al., 2009). Any image cryptosystem is said to be good if the encryption algorithm conceals all attributes of a plain text image and the encrypted image is totally random and highly uncorrelated (Elashry et al., 2009; Kamali et al., 2010). If the encrypted images and plain text images are completely different then their corresponding correlation coefficient must be very low, or very close to zero. The two images are identical and they are in perfect correlation if the correlation coefficient is equal to one.

To test the correlation between two adjoining pixels, the following procedures are performed. First, randomly select 10,000 pairs of two horizontally adjoining pixels from an image and then calculate the correlation.

Coefficient r_{xy} of each pair using the following equations:

$$cov(x,y) = E\big(x - E(x)\big)\big(y - E(y)\big)$$

$$r_{xy} = \frac{cov(x,y)}{sqrtD(x)sqrtD(x)}$$

where x and y are grey-level values of the two adjacent pixels in the image.

Then, the same $E(x) = (\frac{1}{N})\sum_{i=1}^{N} x_i E(x) = (\frac{1}{N})\sum_{i=1}^{N} x_i$ and $D(x) = (\frac{1}{N})\sum_{i=1}^{N}(x_i - E(x)^2)$ $D(x) = (\frac{1}{N})\sum_{i=1}^{N}(x_i - E(x)^2)$ Operations are carried out along the vertical and the diagonal directions, respectively. Table 3 shows the correlation coefficients of the image Lena and its cipher-image, while their correlation distributions are depicted in Fig. 6. The correlation coefficients of the cipher-images are very low, implying that no noticeable correlations exist between the original image and its corresponding cipher-image. Therefore, the proposed algorithm possesses high security against statistical attacks.

Table 2. PSNR value for the encrypted image of Fig. 6, per color channel

	PSNR
Red Channel	8.61 dB
Green Channel	7.051 dB
Blue Channel	6.052 dB
Average PSNR	7.30 dB

Table 3. The correlation coefficients of the image Lena and its cipher-image

	Plain Image	Cipher Image
Horizontal correlation coefficient	0.9715	0.0099
Vertical correlation coefficient	0.9751	0.0144
Diagonal correlation coefficient	0.9534	0.0151

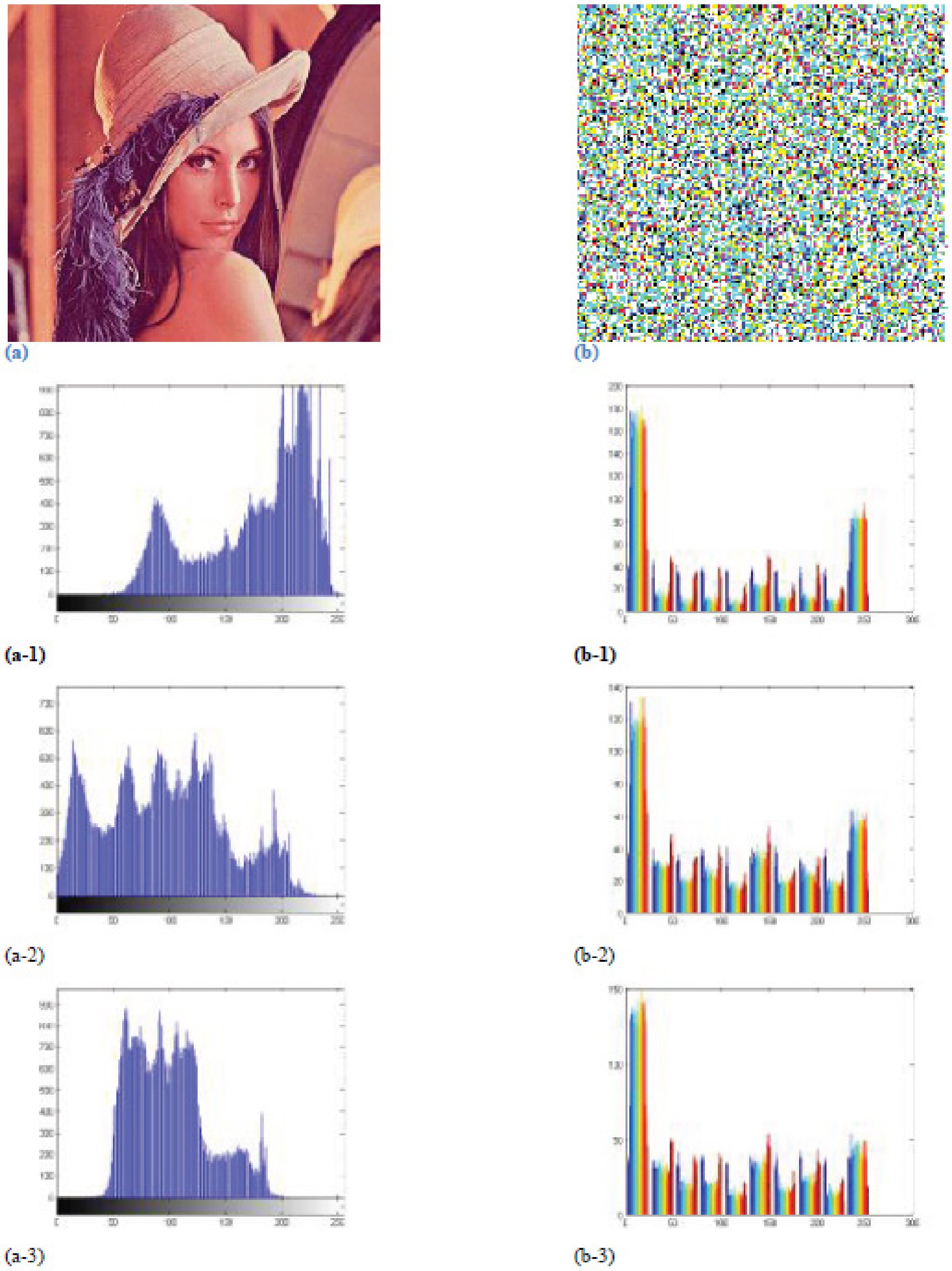
In addition, some experiments are conducted to compare the results of the prosposed method against two other similar works presented in (Eslami, & Ahmadabadi, 2010) verifiable multi-secret sharing scheme (VMSS) and (Zhang, Wang, Zhong, & Yao, 2013) [52] Image encryption scheme based on balanced two-dimensional (B2DIES). The results are shown in Table 4 and Table 5 in the following.

Table 4. The correlation coefficients of the image Lena and its cipher-image provided by the proposed method compared with VMSS and B2DIES in terms of NPCR and UACI

	Proposed scheme			VMSS			B2DIES		
	max(%)	min(%)	average (%)	max(%)	min(%)	average (%)	max(%)	min(%)	average (%)
NPCR	99.4004	99.316	98.3582	99.54	99.54	99.54	99.7736	99.7736	99.7736
UACI	33.9512	33.6461	33.7986	33.49	33.49	33.49	33.4635	33.4635	33.4635

Table 5. The correlation coefficients of the image Lena and its cipher-image provided by the proposed method compared with VMSS and B2DIES in terms of horizontal, vertical, and diagonal correlation coefficient

	Proposed scheme		VMSS		B2DIES	
	Plain Image	Cipher Image	Plain Image	Cipher Image	Plain Image	Cipher Image
Horizontal correlation coefficient	0.9715	0. 9569	0. 9569	0. 00061	0.9173	0.0053
Vertical correlation coefficient	0.9751	0. 8919	0. 8919	0. 00400	0.8425	0.0116
Diagonal correlation coefficient	0.9534	0. 9223	0. 9223	0. 0057	0.7744	−0.0097



(a) (b)

(a-1) (b-1)

(a-2) (b-2)

(a-3) (b-3)

Fig. 5. (a) Secret image, Lena, (a-1) histogram of the red channel of secret image,

(a-2) histogram of the green channel of secret image, (a-3) histogram of the blue channel of secret image, (a) cipher image, (b-1) histogram of the red channel of cipher image, (b-2) histogram of the green channel of cipher image, (b-3) histogram of the blue channel of cipher image

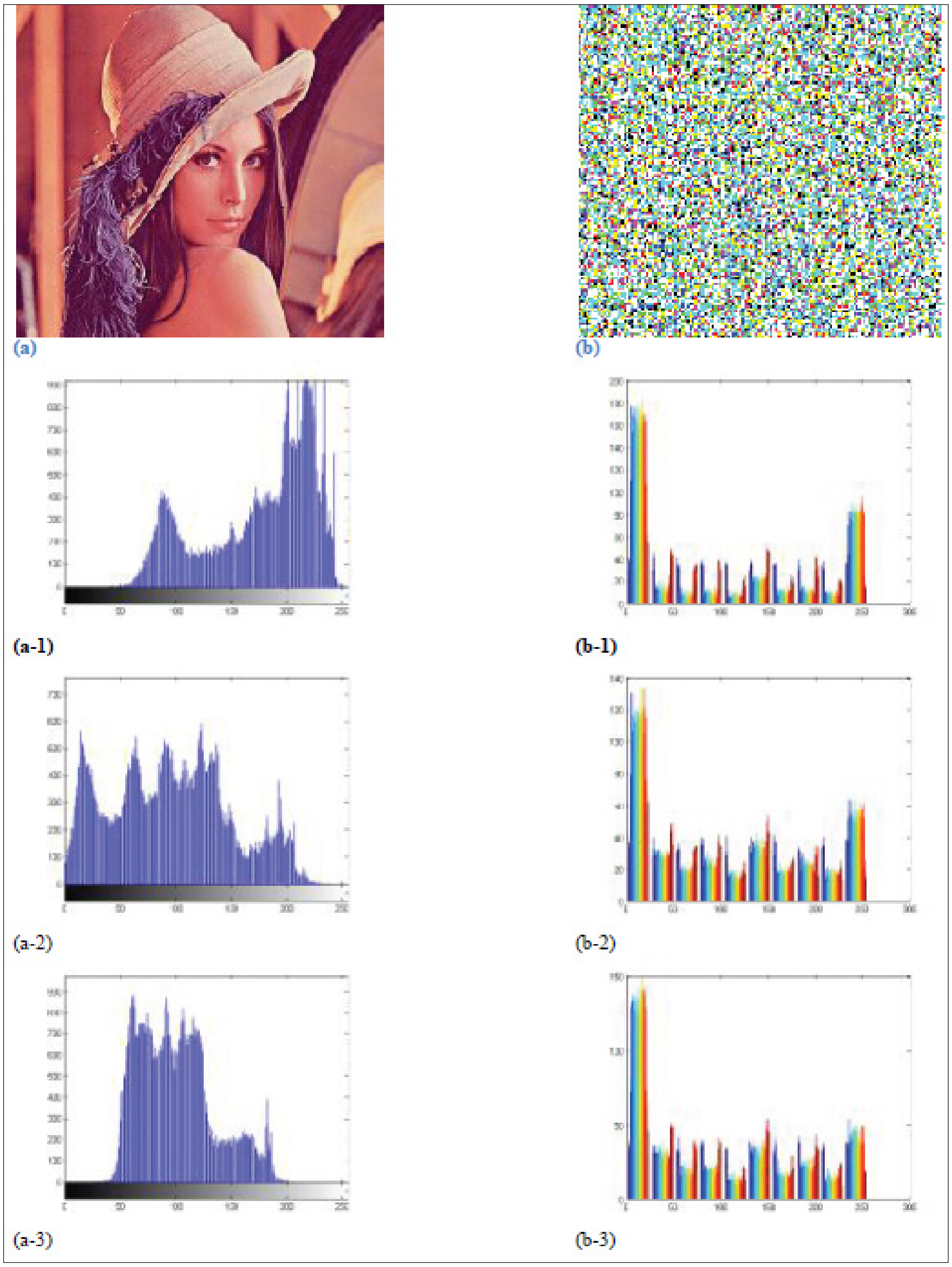


Fig. 6. (a) Secret image, Lena, (a-1) histogram of the red channel of the secret image, (a-2) histogram of the green channel of secret image, (a-3) histogram of the blue channel of the secret image, (b) cipher image, (b-1) histogram of the red channel of cipher image, (b-2) histogram of the green channel of cipher image, (b-3) histogram of the blue channel of cipher image.

Hardware Implementation

The algorithm discussed above is implemented on an lpc1768 board equipped with an ARM Cortex-M3. It is a high-spee d, low-power, and 32-bit processor. The board is shown in Fig. 7.

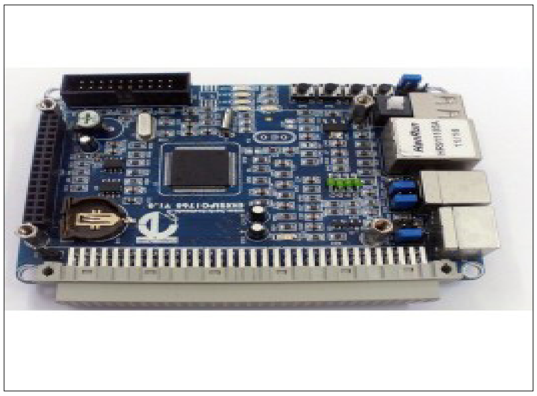


Fig. 6. LPC 1768 board designed for on-board

The specifications of the board is shown in Table 6.

Table 6. The specifications of the board

Processor	Coretex-M3 100MHz
LCD	2.8'' – 320x480 dpi
USB host	1
USB device	1
IC	EEPROM AT24C02
Audio	lm386
LED	1
Interrupt	4 keys
PS2	Keyboard and mouse
LAN	100Mbps
Memory Card	1

The code of the algorithm is implemented on the hardware in the C++ language. The processor is always in a ready state. Whenever the image file – essentially named 'img.bmp' – is input into the board using the USB port, the processor starts to encrypt the image file and the output will be saved as a new file named 'encrypt.bmp'. The decryption process, on the other hand, starts by reading this file and converting it to a valid image named 'decrypt.bmp'.

5- Conclusions

Utilizing particular aspects of cellular automata called "state attractors", a new image encryption method is proposed in this paper. This method exploits the permutation of the image pixels and the replacement of the pixel values. Satisfaction of the properties of confusion and diffusion by the mentioned method is proved, as the characteristics of permutation methods and CA substitution are flexible. The results of security and statistical analysis have proved that the proposed scheme is robust against different kinds of attacks. Moreover, hardware implementation, which is explained in section 5, shows that this cryptosystem can be implemented easily and effectively by simple and cheap hardware. These improvements are observed as a result of using the proposed image encryption method: better statistical properties, better keyspace, lossless decryption, and symmetric private key encryption, and low data expansion, the possibility of encryption when there is more than one image with the use of just one key image and checking data integrity.

Acknowledgement

We are grateful to Islamic Azad University, Quchan branch authorities, for their useful collaboration.

Data availability

The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

References

1-Ahmad, M., Doja, M. N., & Beg, M. M. S. (2021). Security analysis and enhancements of an image cryptosystem based on hyperchaotic system. *Journal of King Saud University-Computer and Information Sciences*, *33*(1), 77-85.

2-Alexopoulos, C., Bourbakis, N. G., & Ioannou, N. (1995). Image encryption method using a class of fractals. *Journal of Electronic Imaging*, *4*(3), 251-259.

3-Bakhshandeh, A., & Eslami, Z. (2013). An authenticated image encryption scheme based on chaotic maps and memory cellular automata. *Optics and Lasers in Engineering*, *51*(6), 665-673.

4-Chai, X., Gan, Z., Yang, K., Chen, Y., & Liu, X. (2017). An image encryption algorithm based on the memristive hyperchaotic system, cellular automata and DNA sequence operations. *Signal Processing: Image Communication*, *52*, 6-19.

5-Das, A., & Adhikari, A. (2010). An efficient multi-use multi-secret sharing scheme based on hash function. *Applied mathematics letters*, *23*(9), 993-996.

6-Del Rey, A. M., Sánchez, G. R., & De La Villa Cuenca, A. (2015). A protocol to encrypt digital images using chaotic maps and memory cellular automata. *Logic Journal of the IGPL*, *23*(3), 485-494.

7-Diab, H., & El-semary, A. M. (2018). Cryptanalysis and improvement of the image cryptosystem reusing permutation matrix dynamically. *Signal Processing*, *148*, 172-192.

8-Enayatifar, R., Sadaei, H. J., Abdullah, A. H., Lee, M., & Isnin, I. F. (2015). A novel chaotic based image encryption using a hybrid model of deoxyribonucleic acid and cellular automata. *Optics and Lasers in Engineering*, *71*, 33-41.

9-Eslami, Z., & Ahmadabadi, J. Z. (2010). A verifiable multi-secret sharing scheme based on cellular automata. *Information Sciences*, *180*(15), 2889-2894.

10-Ferretti, L., Marchetti, M., Andreolini, M., & Colajanni, M. (2018). A symmetric cryptographic scheme for data integrity verification in cloud databases. *Information Sciences*, *422*, 497-515.

11-Guam, P. (1987). Cellular automaton public key cryptosystems. *Complex Systems*, *1*(1).

12-Gutowitz, H. (1993). Cryptography with dynamical systems. In *Cellular Automata and Cooperative Systems* (pp. 237-274). Dordrecht: Springer NetherlandsHanis, S., & Amutha, R. (2018). Double image compression and encryption scheme using logistic mapped convolution and cellular automata. *Multimedia Tools and Applications*, *77*, 6897-6912.

13-Hortensius, P. D., McLeod, R. D., & Card, H. C. (1989). Parallel random number generation for VLSI systems using cellular automata. *IEEE Transactions on Computers*, *38*(10), 1466-1473.

14-Jawad, L. M. (2021). A new scan pattern method for color image encryption based on 3D-Lorenzo chaotic map method. *Multimedia Tools and Applications*, *80*(24), 33297-33312.

15-Lafe, O. (1997). Data compression and encryption using cellular automata transforms. *Engineering Applications of Artificial Intelligence*, *10*(6), 581-591.

16-Li, X., Knipe, J., & Cheng, H. (1997). Image compression and encryption using tree structures. *Pattern RecognitionLetters*, *18*(11-13), 1253-1259.

17-Li, X. W., Kim, S. T., & Wang, Q. H. (2017). Designing three-dimensional cellular automata based video authentication with an optical integral imaging generated memory-distributed watermark. *IEEE Journal of Selected Topics in Signal Processing*, *11*(7), 1200-1212.

18-Li, Z., Peng, C., Tan, W., & Li, L. (2020). A novel chaos-based color image encryption scheme using bit-level permutation. *Symmetry*, *12*(9), 1497.

19-Li, K., Sun, M., Li, L., & Chen, J. (2017). Image encryption algorithms based on non-uniform second-order reversible cellular automata with balanced rules. In *Intelligent Computing Theories and Application: 13th International Conference, ICIC 2017, Liverpool, UK, August 7-10, 2017, Proceedings, Part I 13* (pp. 445-455). Springer International Publishing.

20-Maniccam, S. S., & Bourbakis, N. G. (2004). Image and video encryption using SCAN patterns. *Pattern recognition*, *37*(4),725-737.

21-Nandi, S., Kar, B. K., & Chaudhuri, P. P. (1994). Theory and applications of cellular automata in cryptography. *IEEE Transactions on computers*, *43*(12), 1346-1357.

22-Pieprzyk, J., Seberry, J., & Hardjono, T. (2004). Fundamentals of computer security. *Computing Reviews*, *45*(10), 621-622.

23-Seredynski, F., Bouvry, P., & Zomaya, A. Y. (2004). Cellular automata computations and secret key cryptography. *parallel computing*, *30*(5-6), 753-766.

24-Seredynski, F., Bouvry, P., & Zomaya, A. Y. (2004). Cellular automata computations and secret key cryptography. *parallel computing*, *30*(5-6), 753-766.

25-Sipper, M., & Tomassini, M. (2000). Stream Ciphers with One and Towo-Dimensionals of Cellular Automata. *M. schoenauer et al.(Eds), Parallel Problem Solving From Nature-PPSNVI, LNCS1917, Springer*, 772-731.

26-Talhaoui, M. Z., & Wang, X. (2021). A new fractional one dimensional chaotic map and its application in high-speed image encryption. *Information Sciences*, *550*, 13-26.

27-Vaseghi, B., Mobayen, S., Hashemi, S. S., & Fekih, A. (2021). Fast reaching finite time synchronization approach for chaotic systems with application in medical image encryption. *Ieee Access*, *9*, 25911-25925.

28-Wolfram, S. (1986). Cryptography with cellular automata. In *Advances in Cryptology—CRYPTO'85*

Proceedings 5 (pp. 429-432). Springer Berlin Heidelberg.

29-Wu, X., & Moo, P. W. (1999, June). Joint image/video compression and encryption via high-order conditional entropy coding of wavelet coefficients. In *Proceedings IEEE International Conference on Multimedia Computing and Systems* (Vol. 2, pp. 908-912). IEEE.

30-Yu, S., Lu, J., Leung, H., & Chen, G. (2005, May). N-scroll chaotic attractors from a general jerk circuit. In *2005 IEEE International Symposium on Circuits and Systems (ISCAS)* (pp. 1473-1476). IEEE.

31-Zarepour-Ahmadabadi, J., Shiri-Ahmadabadi, M., & Latif, A. (2018). A cellular automata-based multi-stage secret image sharing scheme. *Multimedia Tools and Applications*, *77*, 24073-24096.

32-Zeng, J., & Wang, C. (2021). A novel hyperchaotic image encryption system based on particle swarm optimization algorithm and cellular automata. *Security and Communication Networks*, *2021*, 1-15.

33-Zhang, X., Wang, C., Zhong, S., & Yao, Q. (2013). Image encryption scheme based on balanced two-dimensional cellular automata. *Mathematical Problems in Engineering*, *2013*.