Modern Applied Science; Vol. 9, No. 13; 2015
ISSN 1913-1844 E-ISSN 1913-1852
Published by Canadian Center of Science and Education

A Novel Dynamic Secret Key Generation for an Efficient Image Encryption Algorithm

Lahieb Mohammed Jawad[1,2] & Ghazali Sulong[1]

[1] UTM-IRDA Digital Media Center (MaGIC-X), Faculty of Computing, Universiti Teknologi Malaysia, Skudai, Johor, Malaysia

[2] Network Engineering Department, College of Information Engineering, Al-Nahrain University, Baghdad, Iraq

Correspondence: Lahieb Mohammed Jawad, UTM-IRDA Digital Media Center (MaGIC-X), Faculty of Computing, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia. Tel: 60-111-629-8170. E-mail: lahieb1978@gmail.com

Received: June 24, 2015 Accepted: September 4, 2015 Online Published: November 30, 2015

doi:10.5539/mas.v9n13p85 URL: http://dx.doi.org/10.5539/mas.v9n13p85

Abstract

Today, the security of digital images is considered a significant essential and a strong secret key plays a major role in the image encryption. In this paper, a novel method for generating dynamic non-linear secret keys for a symmetric block cipher using XOR-operation is proposed. The dynamic non-linear secret keys generation is based on a combination of logistic and piecewise chaotic map methods with a new automatic creation of initial seed values. The automatic initial seed values creation depends on the development of a novel strategy for seeds creation based on sunflower spiral points. The experimental results indicate that the proposed key generator algorithm has the advantage of large key space with a safety protection of brute force attack. Therefore, the performance analysis of image encryption reveals a correlation coefficient of about (-0.0001) and entropy greater than (7.9978). Furthermore, the results show high security for encryption based on strong dynamic secret key properties.

Keywords: secret key generation, randomness, image encryption, sunflower spiral, chaos map

1. Introduction

Recently, high safety is a mandatory requirement in transmitting images over communication networks. As a result of the significant rise in multimedia data usage in computers, there is greater need to verify security and image fidelity. In cryptography, a key is a portion of information that influences the eventual function of a cryptographic algorithm or cipher. In the absence of a key, the algorithm would be to produce a result. A beneficial encryption algorithm must be precise to the cipher keys, and the key space must be sufficiently large to ensure protection against all attacker types such as brute-force attacks (Ephin et al., 2013; Čitavičius & Jonavičius, 2009). The generation of pseudo-random numbers classified as critical subject in a large number of applications such as statistical mechanics, numerical simulations, gaming industry, communication or cryptography. The term “pseudo-random” is used to indicate that the numbers appear to be random are generated from an algorithmic process generator. From a single initial parameter (or seed), the generator will always produce the same pseudo-random sequence. The main advantages of such generators are the rapidity and the repeatability of the sequences and the requirement of less memory for algorithm storage. One interesting way to design such generators is connected to chaos theory. Moreover, during this last decade several pseudo-random number generators have been successfully developed. However, a rigorous analysis is necessary to evaluate the randomness level and the global security of the generator (Saraere et al., 2013).

Today’s world is significantly computerized with high interconnectedness, with increasing focus on the usage of digital chaotic systems that offer opportunities to enhance the security of cryptographic algorithms. The benefits of using chaotic dynamics for security problems lie in their unpredictable nature and in the mathematics-based Theory of Chaos. This theory utilizes several qualitative and quantitative tools, including ergodicity, entropy, expansively, and sensitive reliance on start-up conditions. These tools enable the scrutiny of the disorder randomness produced by the system used (Hua, 2009; BashirAbugharsa et al., 2012). Many researchers have noticed that chaos maps are fit for developing cipher method based on the chaos map fundamental concepts of high sensitivity to the initial parameters and conditions. Chaotic maps are easily impacted by even very small changes in start-up conditions and parameters; it covers a small area of data over the total phase space through iterations. As a result of the sensitive nature of chaotic maps, chaos-based encryption is a novel and effective way of dealing with the persistent challenge of simple but highly significant encryption of secure images (Ahmad et al., 2013). The majority of these novel applications employ chaotic maps as pseudo-random number generators to get a sequence keys, such as that for symmetric encryption. Generators that produce random number are crucial in various fields like the study of statistics, for the purpose of simulation (needed to evaluate performance) or cryptography (Wang et al., 2009).

The nature of the chaotic maps has led cryptographers to produce new algorithms of encryption. The importance of the information encryption is now a popular research direction. Chaos-based schemes deliver a good combination of velocity, complexity, high security, and acceptance efficiency. Nevertheless, most existing protection techniques still suffer from major gabs like slow speed performance, small key size, slow speed performance and weak security (Ahmad et al., 2013). Many researchers combine various chaotic systems with an image encryption system for greater secrecy.

Furthermore, the secret key generator directly determines the security and efficiency of enhancing the image encryption algorithm that is never-ending. In this paper a block-based image encrypting technique for RGB color images is proposed, whereby the matrix secret key is generated using a combination of a non-linear dynamic chaotic system with new automatic initial seed creation. The main idea of this article is to use two nonlinear chaotic methods with new automatic generation of seed values in order to increase the robustness of the generator. To preserve such robustness, we have to avoid collision that may occur with incorrect initial values that will lead to identical series of numbers.

The rest of this paper is organized as follows: section 2 views the preliminary works while, the proposed algorithm is explained in Section 3. The next section describes the security performances and Section 5 explains the comparison. Finally, the conclusion of the presented work is presented in Section 6.

2. Permanent Works

In this section, two main subjects will be explained in detail to clarify the concept of using each one of them. The first part is the chaotic map method while the second one is the sunflower spiral technique.

2.1 Chaotic Map Methods

Chaos Theory is the major researched source in the field of key generation of an image encryption based on its properties of sensitivity of initial value, random behaviors, non-periodic, unpredictable and Correlation properties (Fu et al., 2007). The Chaos system comprises two main phases: confusion and diffusion phases. Diffusion implies extending the impact of a one plaintext digit on many cipher text digits, such that the statistical composition of the plaintext loses clarity. On the other hand, confusion means using transformations that reduce the similarity between cipher and plain text (Parameshachari et al., 2013). Different polynomial dynamical systems are used to generate the chaos sequence such as Piecewise map and logistic map. In this paper the generation of chaotic sequence using the piecewise and logistic maps is studied through the analysis of the bifurcation diagram for each of them. The one dimensional logistic map is used for generating the diffusion stage, which is represented as (Taneja et al., 2012):

$$x_{i+1} = \mu x_i(1 - x_i) \tag{1}$$

where $x_i \in [0, 1]$. Furthermore, x_0 is an initial secret key and μ is the initial parameter of the eq.1 with value $3.57 < \mu < 4$. This map is widely employed by cryptographers to produce a pseudo-random sequence because of its highly sensitivity and chaotic behavior.

Moreover, piecewise linear chaotic maps (called PWLCM) is the main class of chaotic maps which its constantly turbulent for each value of the control parameters of the discrete dynamic systems. The PWLCM is used to generate dynamic secret expressed as (Rhouma et al., 2009):

$$x_{i+1} = \begin{cases} \dfrac{x_i}{m} & 0 \leq x_i < m \\ \dfrac{x_i - m}{1 - m} & m \leq x_i \end{cases} \tag{2}$$

where x_i and m are the iterative value and the system parameter symbolizes the total number of blocks in the image, respectively. To obtain random and non-periodic numbers m is restricted to a range from 0 to 1 (Wang & Liao, 2012).

2.2 Sunflower Spiral Technique

Sunflowers are not just beautiful based on the complex and uniform structure, but it is also a mathematical marvel (Zeng & Pu, 2010). As shown in Figure 1(a), the generating spirals of sunflowers start in the center and there are two series of curves that wind in opposite directions; one starts at the center and stretches out to the petals, and each seed sits at a certain angle from the neighboring seeds. In order to optimally fill the seeds in the center of the flower, the most irrational number is chosen, in other words, the one that is the least well approximated by a fraction. This number is exactly the golden mean. The corresponding angle, the golden angle, is 137.5 degrees. This angle needs to be selected with great precision: as the optimization can be totally destroyed by any changes in degree. When the angle is precisely the golden mean, and only this one, two families of spirals (one in each direction) can be seen and their numbers correspond to the numerator and denominator of one of the fractions which approximates the golden mean. Therefore, the sunflower spiral is a popular and pretty pattern often observable in the way the leaves are arranged or other features in plants, closely connected to the uniform randomness as explained in Figure 1(b).

Mathematically-based on the characteristics of the sunflower spiral, it can be considered a suitable strategy for solving the problems of random number generation with a good uniform distribution characteristic of the sunflower spiral; it can be used for creating automatic seed values with good distribution. The full model proposed by H. Vogel in (1979) defined as, the sunflower spiral is provided by the function S: N → R2, written in polar coordinates as (Segerman, 2010; Ridley, 1982) S(n) = (r(n),θ(n)), where

$$r(n) = \sqrt{n}\,, \tag{3}$$

$$\theta(n) = 2\pi\varphi n \tag{4}$$

where r is the radius while n is the point θ is the golden angle, while φ is the golden ration $\varphi = (\sqrt{5}-1)/2 = \varphi - 1$, finally, the golden ratio is (Segerman, 2010),

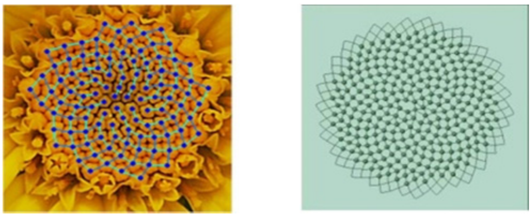
$$\varphi = (\sqrt{5}+1)/2 \tag{5}$$

As explained in Figure 1 (Jones, 2013) [16], the choice of function r (n) gives an equal area packing. However, these polar coordinates can then be converted to Cartesian coordinates to represent the new floret's Cartesian coordinates in the sunflower spiral as shown below (CITA, 2014),

$$x = r * cos(\theta)\ , \text{ and} \tag{6}$$

$$y = r * sin(\theta) \tag{7}$$

Therefore, the plotting points of x and y coordinates can be represented in Cartesian coordinates as shown in Figure 1(b).



a) General view b) Sunflower points

Figure 1. Sunflower architecture (a) General view (b) Sunflower points

3. Proposed Algorithm

Firstly, the plain image is partitioned into n*n blocks where n=16 and the new matrix secret key is dynamically generated based on PWLCM and logistic map with one initial key K0 and develops new automatic initial seed values. Lastly, each block is encrypted using XOR-operations between the block contents and the matrix key and an overview of image encryption and decryption method is explained in Figure 2. The proposed method comprises two main phases: the first phase generates dynamic matrix secret key to be used in the second phase of encrypting each block using XOR-Operation. The first phase consists of two sub-phases to get the final matrix secret key as shown in Figure 3 with the overview block diagram. The first sub-phase creates the automatic initial seeds as explained in a clear example in Figure 4. Moreover, the second sub-phase is used for generating random matrix secret keys dynamically based on the initial seed values from the first sub-phase as shown in Figure 5. These three phases are explained below in more detail.

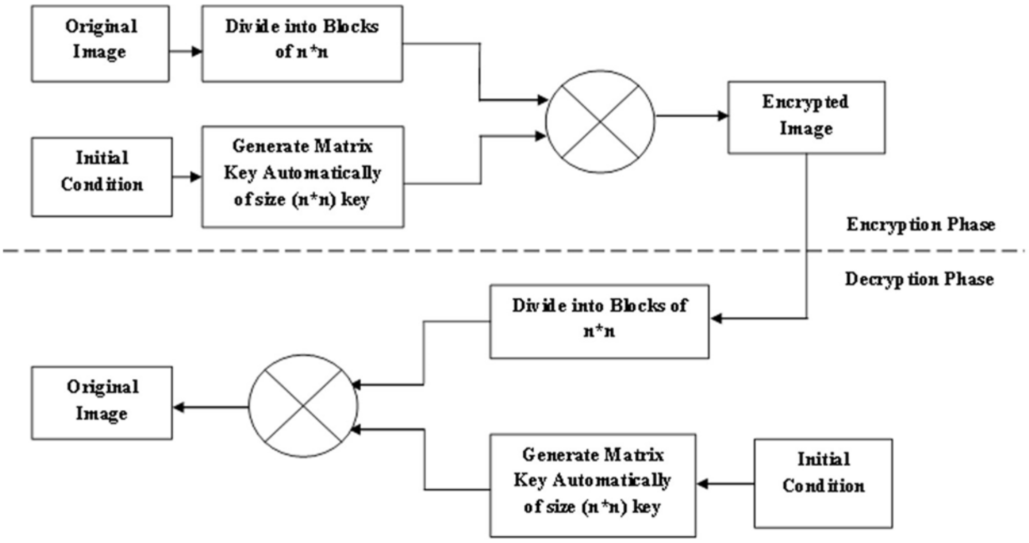

Original Image
Divide into Blocks of n*n
Initial Condition
Generate Matrix Key Automatically of size (n*n) key
Encrypted Image
Encryption Phase
Decryption Phase
Divide into Blocks of n*n
Original Image
Generate Matrix Key Automatically of size (n*n) key
Initial Condition

Figure 2. General block diagram of the image encryption technique

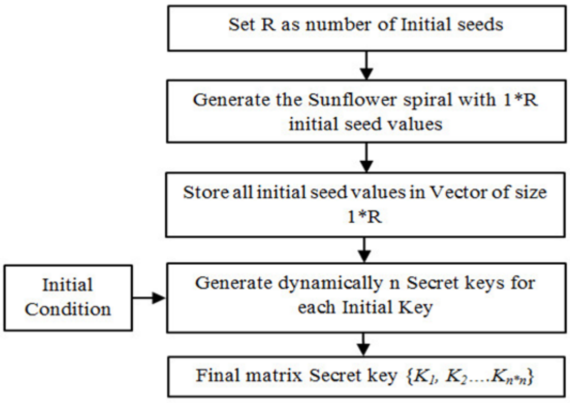

Set R as number of Initial seeds
Generate the Sunflower spiral with 1*R initial seed values
Store all initial seed values in Vector of size 1*R
Initial Condition
Generate dynamically n Secret keys for each Initial Key
Final matrix Secret key $\{K_1, K_2 \ldots K_{n*n}\}$

Figure 3. Block Diagram of Dynamic Secret Key Generation

3.1 New Algorithm for Creating Automatic Initial Seeds

The choice of the starting seed values should not be neglected. Selecting the initial seed for nonlinear dynamic secret key is the first and critical step that determines the strength and the randomness of the sequence for secret key creation. Actually, the initial seed value of the chaotic map method is between [0 and 1]. Therefore, selecting the initial seeds for the chaotic map method is a critical challenge. In this proposed method, a novel strategy is used for selecting this initial seed value automatically using the sunflower spiral points. Based on plotting the Sunflower spiral points for creating the positions of a large number of seeds based on a radius R where R is the total number of seed values. Therefore, Figure 4 gives a good example of creating *n* initial seed where n=100, that is n takes values from 1 to 100, a perfect sunflower head is generated when the distance between points used is 1. Algorithm (1) explains the automatic creation of seed points.

Algorithm1:

Step0: Let R be the total number of initial seed values.

Step1: Determine the radius of sunflower spiral using eq.3

Step2: Evaluate the Golden angle using eq.4 and eq.5

Step3: Determine the position of each seed using eq. 6 and eq.7, respectively.

Step4: Plot the positions of R initial seed.

Step5: Store the X and Y positions of each point in $K_{1..R}$.

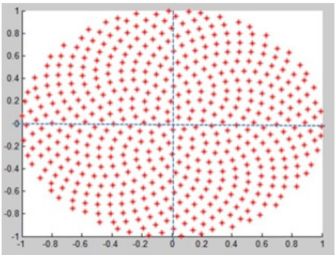


Figure 4. Sunflower spiral points

3.2 Dynamic Secret Keys Generation

In this step the generated matrix secret key will be used in encryption steps. An input parameter for generating matrix secret key is considered with the one seed input secret key (K0). Then, based on the value of K0 a set of secret keys is generated using a combination of logistic and piecewise chaotic map methods. Furthermore, the total number of secret keys remains the same as the number of blocks. The piecewise chaotic map method is used to generate these sets (Wang & Liao, 2012). Figure 5 depicts the block diagram for the dynamic secret key generation. The input is the initial condition μ which ranges between 0 and 1 with 0 < μ ≤ 4 and t is the total number of secret keys generated to obtain the final output {K1, K2… Kn}. Finally, the (R*n) secret keys are generated automatically and will be converted from float values [0...1] to the integer values [0...255]. After that, convert the stream secret keys to matrix of size n*n where n=16.

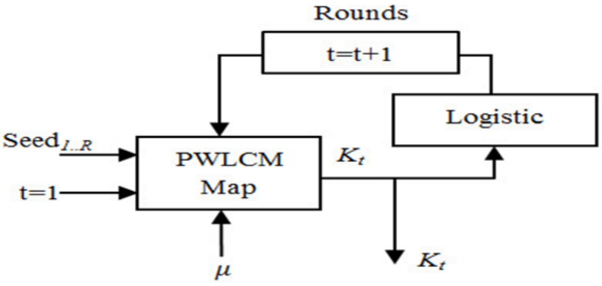


Figure 5. Chaotic-modulation of dynamic secret key generation

3.3 Encryption/Decryption Phase

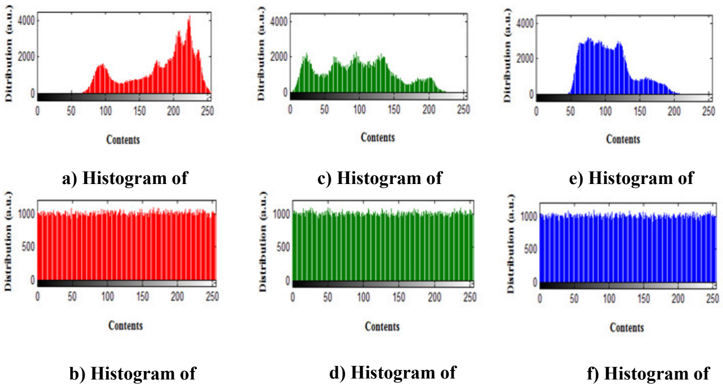
Directly, after the matrix secret key generator of size (n*n) has been initiated, the image is divided into blocks of size (n*n) and encryption is done employing XOR-operation. Regarding the decryption procedure, it is required to use the same external key as explained above in Figure 2.

4. Performance Analysis

Analyzing security involves the identification of the shortcomings of a cryptosystem and retrieval either as a whole or partially of a ciphered image or locating the secret key when the decryption key or the algorithm is unknown. Several experiments are carried out to gauge the efficiency and the effectiveness of our work. We have implemented statistical analysis by computing the histogram, entropy, correlation of two adjacent pixels in the original and encrypted image, the size of key space and the secret key sensitivity. The experiments are evaluated using a standard data set of eight well-known color images accessible in the USC-SIPI Image Database (available online at: http://sipi.usc.edu/database/) with size 512*512. Furthermore, the image that is used to apply this improved algorithm is Lena as a standard image. The results of experimental and statistical analysis explain that our proposed algorithm supports an efficient and safe manner for protection image via network.

4.1 Histogram Analysis

The histogram of an image is considered one of the statistical analyzing features that analyze the distribution of cipher image pixels at individual levels of color intensity. The histograms of ciphered and original image Lena are explained in Figure 6 for red, green and blue channels. To prevent the leakage of information to attackers, it is important to ensure that encrypted and original images do not have any statistical similarities (Huang, 2012). Figure 6 shows that the almost uniform distribution of the histograms of encrypted images which differ considerably from the respective histograms of the original image.



a) Histogram of

c) Histogram of

e) Histogram of

b) Histogram of

d) Histogram of

f) Histogram of

Figure 6. Histograms of original and encrypted image

4.2 Correlation Coefficient Analysis

Lower correlation is a concept to measure the security efficiency of the proposed algorithm based on the fact that the adjacency pixels in an input image are strongly correlated (Usama et al., 2010). Figure 7 utilized correlation the adjacent pixels in the original image and encrypted image. This was followed by randomly selecting 1000 pairs of two adjacent in (vertical, horizontal, and diagonal direction) pixels from an image. Next, with reference to (Wang, & Wang, 2014), the correlation coefficient was measured using the following formula:

$$\mathrm{Cov}(x,y) = \frac{1}{v}\sum_{i=1}^{v}(x_i - E(x))(y_i - E(y)), \tag{8}$$

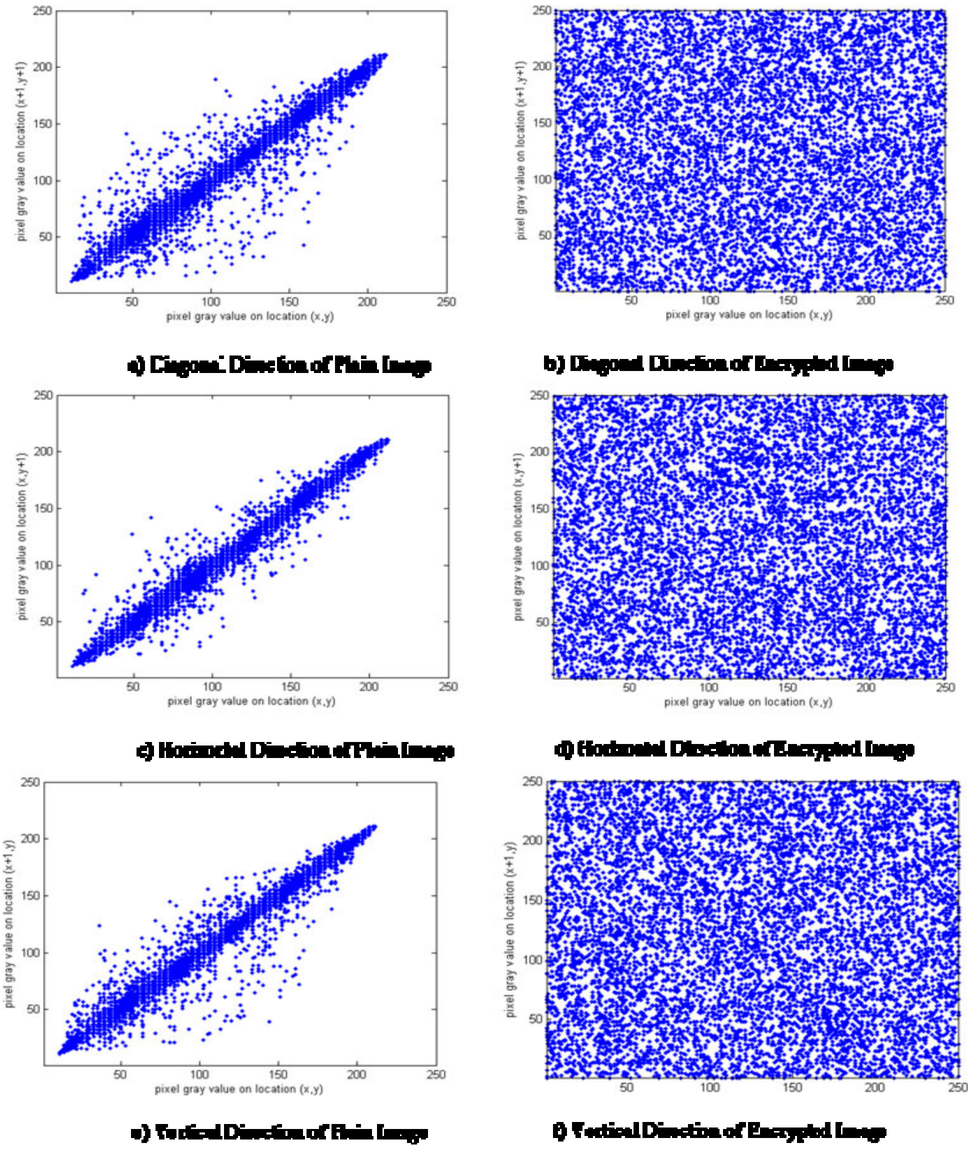
$$CC(x,y) = \frac{Cov(x,y)}{D(x)^{\frac{1}{2}}\, D(y)^{\frac{1}{2}}},$$

with

$$E(x) = \frac{1}{v}\sum_{i=1}^{v}(x_i),$$

$$D(x) = \frac{1}{v}\sum_{i=1}^{v}(x_i - E(x))^2$$

where E(x) and D(y) are the estimated expectation and variance of x, and v is the number of random pairs. Here, Cov(x,y) is the estimated covariance of gray scale values x and y of the two adjacent pixels in the image.



a) Diagonal Direction of Plain Image

b) Diagonal Direction of Encrypted Image

c) Horizontal Direction of Plain Image

d) Horizontal Direction of Encrypted Image

e) Vertical Direction of Plain Image

f) Vertical Direction of Encrypted Image

Figure 7. Correlation Coefficients of two adjacent pixels

In Table 1 and Table 2, the correlation coefficient results are shown and Figure 8 explains the two adjacent pixels are distributed in both vertical and horizontal directions respectively in all dataset original images and their encrypted images. A comparison between the distributed way of cipher and plain images shows it is obvious, that pixels in the original image are of high concentration, whereas the encrypted image pixels are uniformly spread.

As a result, Table 1 provides an excellent presentation of the neighbor pixels in both directions of the original image which are strongly correlated as shown in Figure 8. Correlation coefficients are (+0.0010) and (-0.0012) in horizontal and vertical directions in the encrypted image which is negligible. Thus the analyses of these correlations confirm that the proposed encryption procedure meets zero correlation. Therefore, it could be known that there are no detectable correlations exist between the plain image and its corresponding cipher images.

Table 1. Correlation Coefficient between original and Encrypted Images

	Images Lena	House	Airplane	Sailboat	Baboon	Pepper	Tiffany	Splash
Horizontal CC	+0.0010	+0.0103	-0.0260	+0.0022	-0.0180	+0.0096	-0.0211	+0.0151
Vertical CC	-0.0012	-0.0110	+0.0255	-0.0031	+0.0186	-0.0095	+0.0201	-0.0133

Table 2. Average Correlation Coefficient between original and Encrypted Images

	Images							
	Lena	House	Airplane	Sailboat	Baboon	Pepper	Tiffany	Splash
Avr. CC	-0.0001	-0.00035	-0.00025	-0.00045	+0.0003	+0.00005	-0.0005	+0.0009

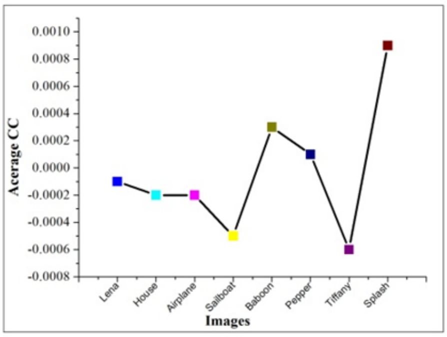


Figure 8. Average CC between original and encrypted images

4.3 Entropy

The entropy is used to quantify the threat-prone rate for retrieving the original image without knowing the key. The value of entropy closer to eight signifies excellent performance. Other values of entropy are considered as the perceptual threat fidelity (Kaur & Singh, 2013). The expression for entropy yields is:

$$H(s)' = \sum_{i=0}^{2n-1} P(s_i) log_2 \frac{1}{P(s_i)}{}' \qquad (9)$$

where P(si) is the probability of the ith event of the image Si and n is the total number of image pixels. A value of entropy, H(s) = 8 corresponds to a truly random source.

Essentially, the security performance of the proposed method in terms of entropy is applied for all dataset images to confirm the protection performances of the proposed algorithm and the calculated value of entropy is listed in Table 3. Moreover, all entropy values in Table 3 are shown in Figure 9 all at the same level, implying that all dataset encrypted images cannot be broken by threat. All the results are generated from the random secret keys that are used for each image and the original image content itself.

Table 3. Entropy Values with key=3.99

	Images							
	Lena	House	Airplane	Sailboat	Baboon	Pepper	Tiffany	Splash
Entropy	7.9979	7.9988	7.9923	7.9988	7.9986	7.9982	7.9962	7.9966

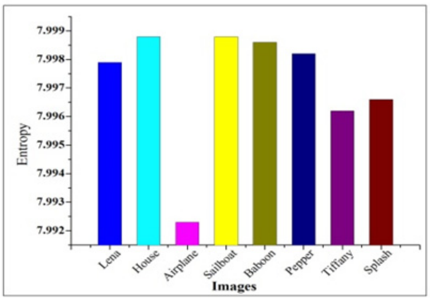


Figure 9. Entropy values of all dataset images

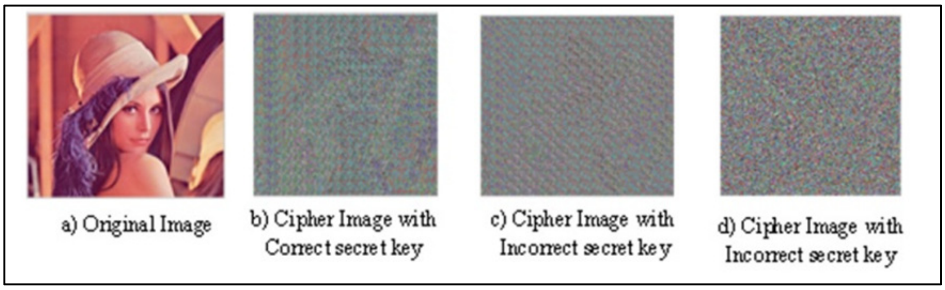
4.4 Key Space Size

An ideal generator of secret key should possess a large key space to ensure that brute-force attacks and exhaustive attack are not possible. It is generally accepted that a key space of size smaller than 2^{128} is not sufficiently secure (Janke, 2002). Here, the key space is constructed from the initial value between (0 and 1) which is used as input of the total number of initial seed values which is needed to generate a dynamic matrix key (as explained in in Section 3.2) therefore, the key space size of the secret key is about 2^{256}. As such, the key space size is sufficiently large to repel brute-force attacks. Such a large space of keys is a crucial necessity, but not adequate enough. Indeed, all the product keys must also be cryptographically strong and uncorrelated.

4.5 Key Sensitivity Analysis

Indeed, a good image encryption algorithm has to be sensitive to the cipher keys and the key space should be appropriately large to repel brute-force attacks. The key space analysis and testing of the proposed algorithm are carefully performed. The total number of pixels in each block is used to generate the secret keys. Therefore, the number of initial keys is varied based on the number of blocks used for encryption. Consequently, the key space depends on the number of pixels for each block. Even for the same initial key the generated key is completely different and nonlinear. The encryption key sensitivity is further tested by comparing the achieved ciphered image by considering a 16-cipher key. The following steps are performed:

1. An original image in Figure 10(a) is ciphered employing the secret key {3.99} called key1 and the resultant image is known as a ciphered image A as indicated in Figure 10(b).
2. The same original image is ciphered by making a minor modify in the secret key, i.e., ={2.95} called key 2 and the resultant image is known as ciphered image B as illustrated in Figure 10(c).
3. Again, the same original image is ciphered by making a slight change in the secret key, i.e., secret key {1.22} called key 2 (modification is made in the last digit) and the resultant image is known as ciphered image C as displayed in Figure 10(d).
4. Finally, a comparison is made of the three ciphered images A, B and C.


a) Original Image

b) Cipher Image with Correct secret key

c) Cipher Image with Incorrect secret key

d) Cipher Image with Incorrect secret key

Figure 10. Key sensitivity of the proposed algorithm for a) original image and b) encrypted image with key1 c) key2 and d) key3

Figure 10 shows the original image together with the three encrypted images. The comparison of the encrypted images by simply observing these images is not easy. Therefore, the correlation between the corresponding pixels of the three encrypted images is calculated using equation 8 except the fact that x and y are the values of corresponding pixels in the two compared encrypted images. The results of the correlation coefficients between the corresponding pixels of the three encrypted images A, B and C are presented in Table 4. It is evident that no correlation exists among the three encrypted images even though they are produced by using slightly different secret keys. This result indicates that the proposed algorithm is very sensitive to any small change in the values of the secret key. So the proposed scheme can resist against brute-force attack.

Table 4. Correlation coefficients between the corresponding pixels of the two encrypted images with key1= {3.99} and different key2 values for all the dataset images

	Images							
	Lena	House	Airplane	Sailboat	Baboon	Pepper	Tiffany	Splash
Avr. CC	+0.00078	+0.0031	-0.00380	+0.00120	+0.00220	-0.00022	-0.00078	+0.00330
Key2	1.6	1.08	0.1	1.88	1.6	1.88	0.33	0.33

4.6 Statistical Analysis of Secret key

Any new secret key must be analyzed against attacks in order to check if the generator cannot be broken. Here, the resistance of the generator against the brute-force attack which is a standard attack that can be used against any secret key. Hence, the purpose of this section is to present the approaches which are used to analyze the qualities of the produced secret key sequences. These qualities are examined properties randomness of each individual sequence and the correlation between multiple sequences. In order to achieve this objective, frequency test is applied and Figure 11 explains the distribution of the secret key sequence sample.

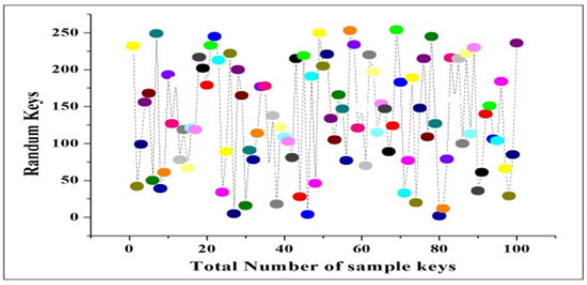


Figure 11. Distribution of the secret key set

Frequency test is one of the basic randomness tests used to evaluate the randomness efficiency. There is no doubt that randomness is the main consideration of a random number generation algorithm. In this test three different seed keys are used to generate three random number generations of secret keys. Each set consists of 100 integer sample keys and these number between 0 and 255 as shown in Figure 12 using three different keys A, B, and C with (1.85, 2.2, and 3.95) respectively.

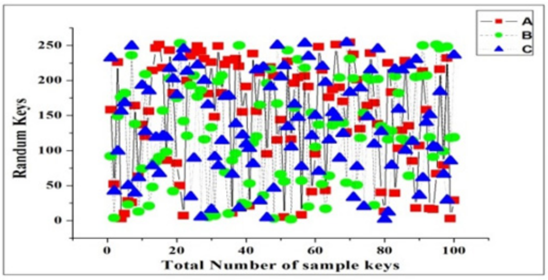


Figure 12. Randomness analysis using three different seed keys

To determine the randomness of all random number generation with various seed keys (0-4), we can find the three curves almost overlap and the variation ranges from (0 - 4) as shown in Table 4. In fact, less variance means the best randomness value (Han et al., 2013). Therefore, we conclude that, when (seed> 0.79), then the random distribution of the random secret key sequences is better.

Results from Table 5 and the graphic representation of the randomness results in Figure 12 are shown to meet the requirements for the uniform distribution. As such, it can be concluded that random key generated by proposing dynamic secret key generation is uniformly distributed.

Table 5. Average Variance for 10 different initial conditions

	Key Numbers									
	1	2	3	4	5	6	7	8	9	10
Initial condition	3.95	3.35	2.5	2.18	1.5	1.2	0.8	0.7	0.5	0.25
Average variance	0.0053	0.0024	0.002	0.0053	0.0057	0.0068	0.0014	595.1	460.33	89.03

5. Comparing the Proposed Method and Others

In this section, the comparison between the proposed algorithm with several algorithms which use the same dataset in gray or color scales. The CC and entropy of the ciphered images are computed for each algorithm and the results are provided in Table 6 and Table 7 that clearly display the superior security level of the proposed algorithm compared to others. The possibility of threat is minimal and no one can break the cipher without knowing the secret key. Data in Table 6 and Table 7 exhibit the achievement of highest entropy and lowest correlation coefficients for all CC directions using the proposed technique at different color scales compared to several other techniques. This comparison reveals that the proposed technique is able to achieve the best security performance.

Table 6. Comparison of CC values of proposed technique with other algorithms

	Image Encryption Algorithms			
	Proposed algorithm (ours)	(Saberi., 2014)	(Abugharsa & Almangush, 2011)	(Faridnia & Fae'z, 2010)
Horizontal CC	-0.01470	+0.00560	-0.00780	+0.02890
Vertical CC	+0.01450	-0.00590	-0.05550	+0.01910
Average CC	-0.00010	-0.00015	-0.03165	+0.02400

Table 7. Comparison of entropy values of proposed technique with other algorithms

	Image Encryption Algorithms			
	Proposed algorithm (ours)	(Saberi., 2014)	(Abugharsa & Almangush, 2011)	(Faridnia & Fae'z, 2010)
Scale	Color	Color	Color	Gray
Entropy	7.9979	7.9919	7.9919	7.9911

6. Conclusion

We conclude that a secret key plays a very important role in encryption security, so that a good key generation unit is required. This paper, proposed a new dynamic random number generation employing a combination of two chaotic map methods based on the properties of chaos functions and the possibility of creating very long length keys. Based on the key's large space in the combined chaos functions, this technique is very robust. Moreover, it is very sensitive to even minute modifications in key so even with knowing the key's approximate values it is still not possible for the attacker to break the cipher. In addition it creates automatic seed values based on sunflower spiral points. The advantages of the generator are: high sensitivity to initial seed values, high level of randomness and good throughput. Although what is presented is the dynamic secret key generator, the proposal has focused on an image protection, but it can be applied extensively in other information security fields with excellent results. Our proposal enables sending an initial condition to another party over the open network. The varying space sized block secret keys are randomly produced with one initial condition. It is further demonstrated that these dynamic matrix secret key are responsible for increasing the security level based on large key space size with better randomness distribution.

Acknowledgments

Lahieb Mohammed Jawad is grateful to the Ministry of Higher Education and Scientific Research, Iraq for providing sponsorship to continue her PhD.

References

Abugharsa, A. B., & Almangush, H. (2011). A new image encryption approach using block-based on shifted algorithm. *International Journal of Computer Science and Network Security (IJCSNS), 11*(12), 123-130.

Ahmad, M., Chugh, H., Goel, A., & Singla, P. (2013). A chaos based method for efficient cryptographic S-box design. In Security in Computing and Communications (pp. 130-137). Springer Berlin Heidelberg.

BashirAbugharsa, A., Samad Bin Hasan Basari, A., & Almangush, H. (2012). A new image encryption approach using the integration of a shifting technique and the AES algorithm. *International Journal of Computer*

Applications, 42(9), 36-45.

CITA Capstone Project (2014). Retrieved May 2, 2014, from http://msrourk.wordpress.com/2014/05

Čitavičius, A., & Jonavičius, A. (2009). An image encryption using pseudo random bit generator based on a non-linear dynamic chaotic system. *WSEAS TRANSACTIONS on COMMUNICATIONS, 8*(9), 1022-1031.

Ephin, M., Vasanthi, N. A., & Joy, J. A. (2013). *Survey of Chaos based Image encryption and decryption techniques. In International Journal of Computer Applications.* Proceedings of the 2rd International Conference on Amrita International Conference of Women in Computing 2, pp. 1-5.

Faridnia, S. E., & Fae'z, K. (2010). Image encryption through using chaotic function and graph. Proc. of the 1st Int'l Conference on Computer Vision and Graphics (pp. 352-359). Warsaw, Poland, 2, Springer Berlin Heidelberg.

Fu, C., Zhang, Z. C., Chen, Y., & Wang, X. W. (2007). An Improved Chaos-Based Image Encryption Scheme. In Computational Science–ICCS 2007 (pp. 575-582). Springer Berlin Heidelberg.

Han, Ch., Wang, Y., & Liu, Y. (2013). Two Improved Pseudo-Random Number Generation Algorithm Based on the Logic Map. *Research Journal of Applied Sciences, Engineering and Technology, 5*(6), 2174-2179.

Huang, X. (2012). Image encryption algorithm using chaotic Chebyshev generator. *Nonlinear Dynamics, 67*(4), 2411-2417.

Hua, J., & Shasha, Z. (2009). The Network Identity Authentication System Based on Iris Feature Identification. *Modern Applied Science, 3*(5), 127-130.

Janke, W. (2002). Pseudo random numbers: Generation and quality checks. Quantum Simulations of Complex Many-Body Systems: From Theory to Algorithms.–John von Neumann Institute for Computing.–Jülich.–2002.–NIC Series, 10, 447-458.

Jones, C. (2013). Magical Sunflowers-Fibonacci Spiral and Heliotropism. Retrieved August 14, 2013, from https://thegardendiaries.wordpress. com /2013/08/14/magical-sunflowers

Kaur, R., & Singh, E. K. (2013). Comparative analysis and implementation of image encryption algorithms. *International Journal of Computer Science and Mobile Computing (IJCSMC), 2*(4), 170-176.

Parameshachari, B. D., Soyjaudah, K. S., & Chaitanyakumar, M. V. (2013). A study on different techniques for security of an image. *International Journal of Recent Technology and Engineering (IJRTE), 1*(6), 14–19.

Rhouma, R., Arroyo, D., & Belghith, S. (2009, March). A new color image cryptosystem based on a piecewise linear chaotic map. In 6th International Multi-Conference on Systems, Signals and Devices, Djerba, Tunisia, Mar. 23-26 (No. 4956666, pp. 1-6).

Ridley, J. N. (1982). Packing efficiency in sunflower heads. *Mathematical Biosciences, 58*(1), 129-139.

Saberi., M., Mohammad, D., Rahim, M. S. M., & Yaghobi, M. (2014). Using 3-cell chaotic map for image encryption based on biological operations. *Nonlinear Dynamics, 75*(3), 407-416.

Saraereh, O. A., Alsafasfeh, Q., & Arfoa, A. (2013). Improving a New Logistic Map as a New Chaotic Algorithm for Image Encryption. *Modern Applied Science, 7*(12), 24. http://dx.doi.org/ 10.5539/mas.v7n12p24

Segerman, H. (2010, July). The Sunflower Spiral and the Fibonacci Metric. In Proceedings of Bridges 2010: Mathematics, Music, Art, Architecture, Culture (pp. 483-486). Tessellations Publishing.

Taneja, N., Raman, B., & Gupta, I. (2012). Combinational domain encryption for still visual data. *Multimedia Tools and Applications, 59*(3), 775-793.

Usama, M., Khan, M. K., Alghathbar, K., & Lee, C. (2010). Chaos-based secure satellite imagery cryptosystem. *Computers & Mathematics with Applications, 60*(2), 326-337.

Vogel, H. (1979). A better way to construct the sunflower head. *Mathematical biosciences, 44*(3), 179-189.

Wang, L., & Liao, X. (2012). A novel image encryption approach based on chaotic piecewise map. *Journal of Theoretical Physics and Cryptography, 1,* 37-40.

Wang, Q., Guyeux, C., & Bahi, J. M. (2009, August). A novel pseudo-random number generator based on discrete chaotic iterations. First International Conference on an Evolving Internet (pp. 71-76). IEEE.

Wang, X., & Wang, Q. (2014). A novel image encryption algorithm based on dynamic S-boxes constructed by chaos. *Nonlinear Dynamics, 75*(3), 567-576.

Zeng, L., & Pu, Q. (2010, August). *Interactive flower modeling based on phyllotactic pattern.* In 2010 Sixth International Conference on Natural Computation (ICNC), 8, 4075-4079. IEEE.

Copyrights