Kirkuk University Journal /Scientific Studies (KUJSS)
Volume 13, Issue 1, October 2018, pp. (104-117)
ISSN 1992 – 0849

Color Image Encryption Using Hybrid Method of Fractal-Based Key and Private XOR Key

Nevart A. Minas[1], Faten H. MohammedSediq[2], Adnan Ibrahem Salih[3]

[1,2] Computer Systems Department, Technical Institute of Kirkuk, Northern Technical University, Kirkuk, Iraq.
[3] Computer Department, Collage of Science, Kirkuk University, Kirkuk, Iraq.

Nev_al122@yahoo.com[1], fatenhello@yahoo.com[2], adnan_alezzi@uokirkuk.edu[3]

ABSTRACT

The main idea of image encryption is to ensure secure transmitting it in the open network by transforming the image to not understandable form so that unauthorized person cannot decrypt and see it. This paper introduces a hybrid image encryption method based on the details of the Mandelbrot set fractals and the private XOR operation. Due to the random nature of fractal images, they can be used as a strong key for encryption. The strength of the key caused by using three random parameters as a secret keys (two of them are the zooming position values x & y, and the third is number of zooming times form that position) used to create the fractal-based key image, and they can generate large number of random fractal images. To make the secret image more secure, the original image is encrypted by applying XOR bit operation on each adjacent pixels of the image. At last the resulted image is combined with the previously obtained fractal-based key image using a predefined private XOR table in a complex order. To evaluate the performance of the system, the correlation coefficients, visual test using histogram analysis and different secret keys were used. All tested results proved that the proposed method is promising and effective to be used in image encryption fields.

Keywords: Image Encryption, Cipher, XOR, Fractal Geometry, Mandelbrot.

تشفير الصور الملونة بأستخدام طريقة هجينة من المفتاح المبني على الكسورومفتاح أو الحصرية الخاص

نفارت الياس يوسف[1], فاتن حسن محمد صديق[2], عدنان ابراهيم صالح[3]

[1,2] قسم انظمة الحاسوب، المعهد التقني كركوك،الجامعة التقنية الشمالية ، كركوك، العراق.
[3] قسم الحاسبات، كلية العلوم، جامعة كركوك، كركوك، العراق.

Nev_al122@yahoo.com[1], fatenhello@yahoo.com[2],
adnan_alezzi@uokirkuk.edu[3]

الملخص

الفكرة الاساسية من تشفير الصور هو ضمان نقل أمن للصور في الشبكات المفتوحة، وذلك بتحويل الصورة الى صيغة غير مفهومة بحيث لايستطيع الاشخاص غير المخولين فكها ورؤيتها. استخدم البحث طريقة هجينة لتشفير الصورة بناءآ على تفاصيل الكسور من مجموعة (Mandelbrot) وعملية أو الحصرية الخاصة.
ساعدت الصفة العشوائية للصور الكسورية على امكانية استخدامها كمفتاح قوي للتشفير, وتكمن قوة المفتاح من خلال اعتماده على ثلاث متغيرات عشوائية كمفاتيح سرية (متغيران لموقع نقطة التكبير متمثلة بالاحداثيات (x) و (y) والمتغير الثالث يمثل عدد مرات التكبير من تلك النقطة) التي تستخدم لتكوين صورة المفتاح الكسوري الاساسي حيث بالامكان تكوين عدد كبير من الصور الكسورية العشوائية. لزيادة امنية الصورة الاصلية تشفر بتطبيق عملية أو الحصرية لكل نقطتان متجاورتان فيها، واخيرا تربط الصورة الاصلية الناتجة مع صورة المفتاح الكسوري الاساسي باستخدام جدول خاص لدالة أو الحصرية معد مسبقا بتنظيم معقد. تم تقييم أداء عمل البحث المقترح باستخدام كل من معامل الارتباط، الاختبار باستخدام تحليل المخططات واستخدام مفاتيح سرية مختلفة واظهرت نتائج الاختبارات ان النظرية المقترحة هي طريقة واعدة وكفوءة في مجال تشفير الصور

الكلمات المفتاحية : تشفير الصور، الدالة XOR، الدالة الكسورية، Mandelbrot، معامل الارتباط.

.

1. Introduction

Nowadays huge amounts of digital visual data are exchanged over various sorts of networks, these visual data contain confidential information's. As a consequence, many techniques are required to provide security, like integrity, privacy, and authentication suitable for these data types [1]. Digital cryptosystems are divided into two types according to key distribution: symmetric and asymmetric key. Symmetric-key cryptosystems use a secret key for encryption and decryption. They are appropriate for processing large amounts of high speed data. The length of the symmetric cipher keys are usually of range (128 to 256) bits. Asymmetric key cryptography also called, Public key cryptography, uses a pair of keys: public and private keys. Cryptography technique needs some algorithm for encryption of data. Nowadays a huge sensitive information is transferred over the Internet that need to ensure its security and safety. Images are an important part of these information, therefore it is very important to protect them from unauthorized access [2].

At present, many image encryption algorithms are produced [3-6] trying to convert the image to unintelligible form that is hard to understand. The term “Fractal” was proposed by Mandelbrot in 1967 when studying patterns on England. The fractal image can be efficiently used as a key in the encryption methods because it is very difficult to break this type of keys against attacks. Generation of fractal image is changed in large amounts when a small change occurred in any of the parameters of fractal image. So it can be used as a secure encryption key. Several papers have been proposed on image encryption using fractal key for image encryption [7-10]. To study the differential attacks, the characteristics between the input image and the output ciphered image have to be meaningful. Strong encryption systems must be sensitive to minimal changes in the input to produce a totally different output [8]. In this paper, fractal image of Mandelbrot Set is used as a strong symmetric key to encrypt the source image using the XOR operation.

2. Fractal Geometry

Fractal geometry is a repeated pattern in images, that means, each image or object has this feature is divided into small sections. Each of these sections are a small copy of the original pattern [11]. Fractal Image is a self-similar object [12] obtained by repeating a section at various scales of enlargement possessing high variations. It can be generated by a mathematical equation that is iterated for a finite number of times. The choice of the fractal image depends on the level of its details and colors. Many resources are available on the Internet to generate fractals [13], most popular fractals can be obtained through Iterated Function Systems (IFS). IFS have attracted a lot of attention because of their computational simplicity, efficiency, and great ability to randomly reproduce natural and complex patterns [14].

A fractal object has two basic characteristics: infinite detail at every point and a certain self-similarity between the object parts and the overall features of the object. Fractal images are very sensitive to initial values, a big change is taking place when a small change occurred in one of the fractal parameters during its generation. Its curves have non-integer dimensions which are between 1 and 2 [7]. The best known fractal is Mandelbrot set. Mandelbrot set is a set of points in the complex plane. The complex plane is a two-dimensional space with the a vertical imaginary axis, and a horizontal real axis. This set is beauty and complex and derived from some simple definitions (equ. 1). It is defined as a set of points C0 in the complex plane, for which the sequence is defined by the iteration, such that the infinite sequence C_0, C_1, …, C_n, remains bounded, where:

$$C_{0=}\ C_{0,} \qquad C_{n+1}=C_n^2+C_0 \quad \text{for } n=0,\ 1,\ 2,\ 3,\ \ldots \qquad (1)$$

Most of techniques uses the Mandelbrot fractal curve as an encryption key. A typical Mandelbrot set [15] shown in Fig. (1) can be defined by the following function [16]:

$$f : z_{n+1} \longleftarrow z_n^2+c \qquad (2)$$

It consists of complex values “C-values” and the sequence of the repeated combination of the function with itself do not reach infinity Fig. (1) shows some set of fractals based on the Mandelbrot Set [11].

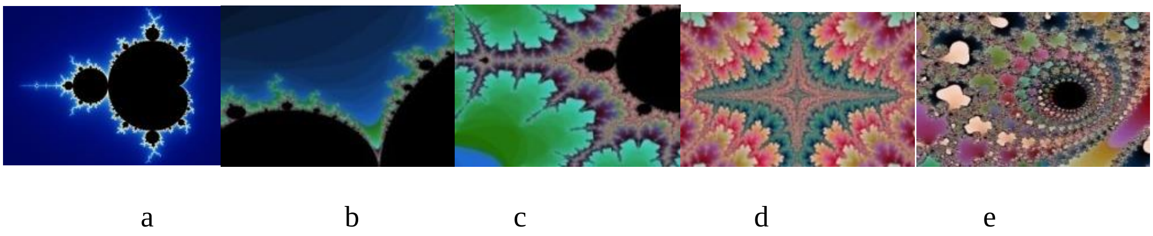


Fig. (1): Set of different fractals

(a) The default Mandelbrot, (b-e) generated fractals from the default(a)

The Mandelbrot Set is an infinitely detailed mathematical generated by the formula in equation (1) as follows: Given the coordinates of a pixel (*x* and *y*), recorded as (*j* & *k*). The *x* is repeatedly replace with (x^2-y^2+j and y) is replaced with (2xy+k). If the numbers grown was little huge, count the number of how many this has been repeated. If the point (x^2+y^2) passed the value 4, stop the iterating and set the color based on this number [17]. Each iteration represents a color (0 - 255), if the color passed the maximum it will return to (0). *The* default value of iteration parameter is (100), when increased to (1000), it will lead to zoom-in the image. Large values are preferred for better resolution of the Mandelbrot set, while the smaller values require less computation time [18].

3. Proposed Encryption Method

The proposed image encryption scheme is presented using C# language. It depends on three methods: The fractal geometry, the Boolean XOR operation, and the private XOR operation. The source image can be of any type (.bmp or .jpeg, or any other type) The algorithm is done on four phases as follows:

A.Encryption of the Original Image with XOR Operation:

To get more secure image, the source image is also encrypted by applying the Boolean XOR bit operation on every two adjacent pixels storing the results in the first pixel creating the first step cipher image (Cipher1).

B.Create the Fractal Cryptographic key image:

To generate any fractal image some parameters are needed as a secret keys. Three keys are used here for encryption operation which must be secret between the sender and the receiver of the image. The first two keys are the zooming position (x & y), which their values must not be more than the original image dimensions, the third is the number of zooming times, it preferred to be less than (10) times to avoid extra zooming that cause to move the position to an empty area or less detailed area that leads to weak fractal images. In this work, the fractal rectangle edges are set to (-2, 2, 2, -2) that determined the left, right, upper, and down edges of the drawing rectangle simultaneously. The parameter that represents the fractal iterations is set to 1000.

The Mandelbrot fractal image contain wide empty space which is not efficient in its default state Fig. (1)-a to be used in the encryption operation, so another image is generated from the default image to be the base fractal key. When starting encryption, the system is automatically generates this image by repeatedly calling the fractal generation program from different selected positions many times to obtain a new detailed fractal image which will be used as a base image to encrypt all images in the system. From this image and using the entered three secret keys, the new fractal key is obtained. Instead of sending the whole secret image as a key, only the three parameters are needed, and they will be used to create the secret key image needed in the encoding and decoding stages.

The secret key space in the proposed method is $2^{(N \times M \times I)}$ considering that the image size is *(N*M)* and the zooming iteration number is *(I)*. This means we can generate one of thousands different images at each time the system is used which increases the secrecy of the system.

C. Private XOR Generation

The XOR operation is frequently used in most of image encryption systems, so it is easy to be attacked and get the original image. To get more secure image, a new predefined private XOR table is generated instead of the ordinary XOR operation. The private XOR values must achieve the following conditions:

1. All values are unique and with range (0 - 15).
2. The sum of values on each row and column is equal to 120.
3. Values in the upper triangle must be the same as the lower triangle values.

Tabe (1) - a and b shows the difference between the ordinary XOR and the Private XOR tables. Each table consists of two dimensional arrays (16x16) of 4-bit (0-15) values managed with XOR and private XOR operation, and can encrypt only half byte(4-bit), so it have to be used twice to encrypt each byte (8-bit) of the image.

D.Combine The Source Image With The Fractal Key:

The encrypted image obtained from the first phase (Cipher1) is combined with the fractal image obtained from the second phase using the private XOR table to obtain the second step Cipher image (Cipher2). Fig. (2) show the designed system in C# language displaying the base fractal image and the three secret keys, and the new generated Mandelbrot fractal key. Fig. (3) shows the block diagram of the proposed encryption algorithm.

Table (1): The difference between ordinary XOR and the proposed Private XOR operation

(a) The ordinary XOR table.

4-bit values (0-15)

4-bit values (0-15)

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	0	3	2	5	4	7	6	9	8	11	10	13	12	15	14
2	2	3	0	1	6	7	4	5	10	11	8	9	14	15	12	13
3	3	2	1	0	7	6	5	4	11	10	9	8	15	14	13	12
4	4	5	6	7	0	1	2	3	12	13	14	15	8	9	10	11
5	5	4	7	6	1	0	3	2	13	12	15	14	9	8	11	10
6	6	7	4	5	2	3	0	1	14	15	12	13	10	11	8	9
7	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8
8	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7
9	9	8	11	10	13	12	15	14	1	0	3	2	5	4	7	6
10	10	11	8	9	14	15	12	13	2	3	0	1	6	7	4	5
11	11	10	9	8	15	14	13	12	3	2	1	0	7	6	5	4
12	12	13	14	15	8	9	10	11	4	5	6	7	0	1	2	3
13	13	12	15	14	9	8	11	10	5	4	7	6	1	0	3	2
14	14	15	12	13	10	11	8	9	6	7	4	5	2	3	0	1
15	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

(b) The proposed Private XOR table

4-bit values (0-15)

4-bit values (0-15)

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	7	8	3	2	14	12	6	0	1	9	15	11	5	13	4	10
1	8	9	2	14	6	15	4	7	0	1	10	13	12	11	3	5
2	3	2	1	0	13	9	14	15	12	5	11	10	8	4	6	7
3	2	14	0	9	10	11	13	12	15	3	4	5	7	6	1	8
4	14	6	13	10	7	8	1	4	5	15	3	12	11	2	0	9
5	12	15	9	11	8	5	10	13	4	2	6	3	0	7	14	1
6	6	4	14	13	1	10	0	11	8	12	5	7	9	3	2	15
7	0	7	15	12	4	13	11	1	14	10	9	6	3	5	8	2
8	1	0	12	15	5	4	8	14	6	11	13	9	2	10	7	3
9	9	1	5	3	15	2	12	10	11	0	7	8	6	14	13	4
10	15	10	11	4	3	6	5	9	13	7	1	2	14	8	12	0
11	11	13	10	5	12	3	7	6	9	8	2	0	4	1	15	14
12	5	12	8	7	11	0	9	3	2	6	14	4	1	15	10	13
13	13	11	4	6	2	7	3	5	10	14	8	1	15	0	9	12
14	4	3	6	1	0	14	2	8	7	13	12	15	10	9	5	11
15	10	5	7	8	9	1	5	2	3	4	0	14	13	12	11	6

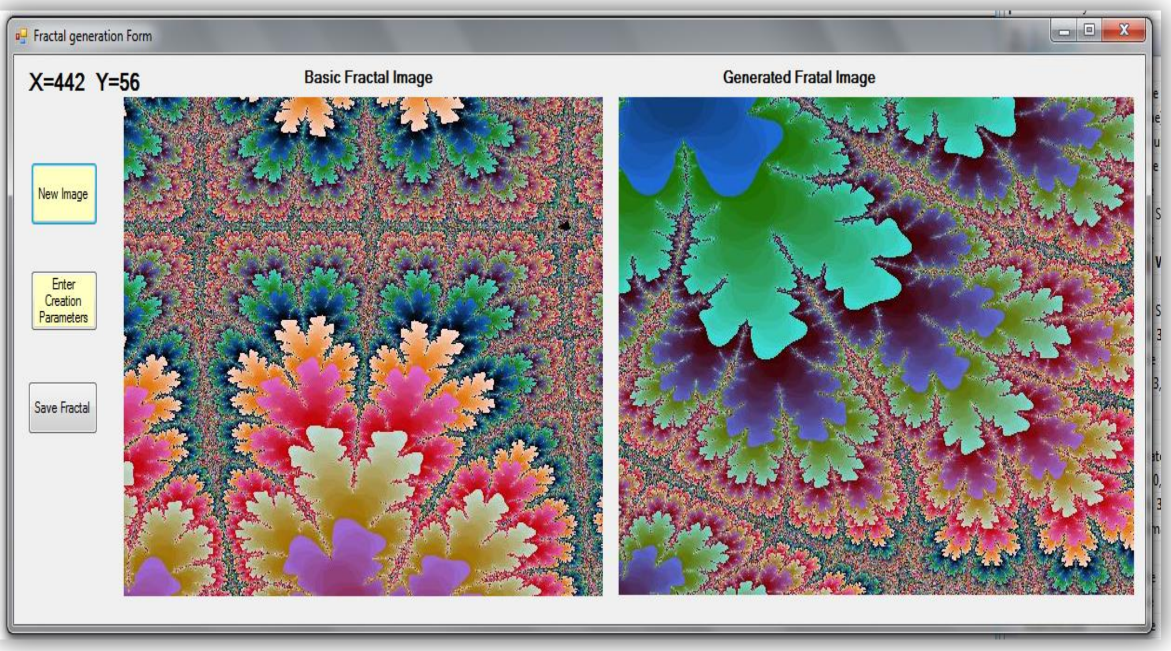


Fig. (2): The fractal generation system in C# language showing the base and generated fractal images.

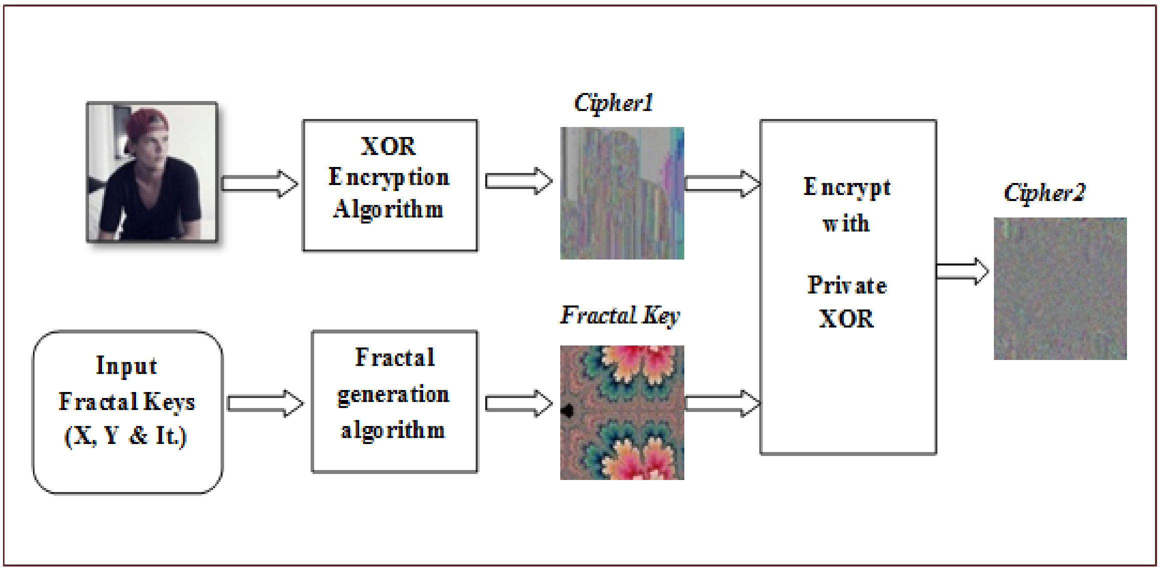


Fig. (3): The proposed Encryption block diagram.

The Encryption steps are summarized by the following algorithm:

step1. Input: Original colored image.

step2. Generation of the Private XOR table.

step3. Create Cipher-1 (Encrypting each adjacent pixels of the Original Image with the ordinary XOR operation).

step4. Input the Three secret keys: zooming dimensions (X) and (Y) and the zooming value.

step5. Generate the Fractal Cryptographic key image using Fractal geometry equations (1-2).

step6. Create Cipher-2 by encryption of the source image with the Fractal key image (i.e. Cipher-1 with the fractal key in step 5) using private XOR table.

step7. Output: Final encrypted image.

The decryption algorithm to get back the original image is the same as the encryption algorithm but in reverse steps. Firstly the same secret keys are used to create the fractal key image as it was in the encryption process. Secondly, this image is combined with the encrypted image (Cipher2) using the private XOR to get the (Cipher1) image. At last, the XOR bit operation is applied on each adjacent pixels in (Cipher1) to get back the original image. The proposed algorithm can be classified as lossless, so the original and decrypted images are identical.

4. Experimental Results

Different encryption keys were applied on different source images (Boy, Room, and Group), with different input keys, that means different fractal images can be generated from the base fractal. Different fractal images that generated from many combinations of input keys were tested in the proposed system. Fig. (4) shows the Source colored image, the first phase was the encrypted image (Cipher1) ,and the final phase was the encrypted image(Cipher2) that obtained from combining (Cipher1) with the base fractal key using the Private XOR operation. The first encrypted image was visually good encrypted, but it might be attacked easily because of using the ordinary XOR, so the final encrypted image was provided with high security level and it is robust against any attack.

The histogram is a visual test showing the different intensity values for number of pixels in an image. Fig. (5) shows three histograms for each tested image: the original image histogram and the two phases of the encryption histograms. The second phase is more resistant to all security and statistical attacks.

Another visual test is applied using the image (Boy) with different fractal keys as examples to compare the effect of different fractals on the final encrypted images as shown in Fig. (6). The first example is the base fractal image, the second is generated from the position (439,241) with repeated 3 times from this position, and the third is generated from the position (332,196) and repeated 5 times from this position. The histograms of the three fractal keys showed a high correlation for all encrypted images, that means at any selected combination of secret keys, the correlation will be high.

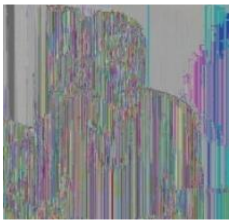
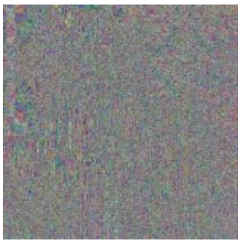
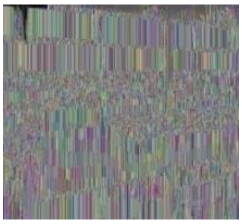
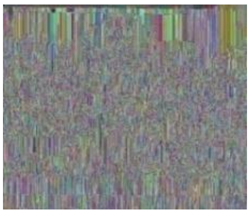
Original Image	Cipher1 (Encrypted original image using XOR)	Cipher2 (Encrypted with fractal key and Private XOR)
		
		
		

Fig. (4): Source and encrypted images (Cipher1& Cipher2)

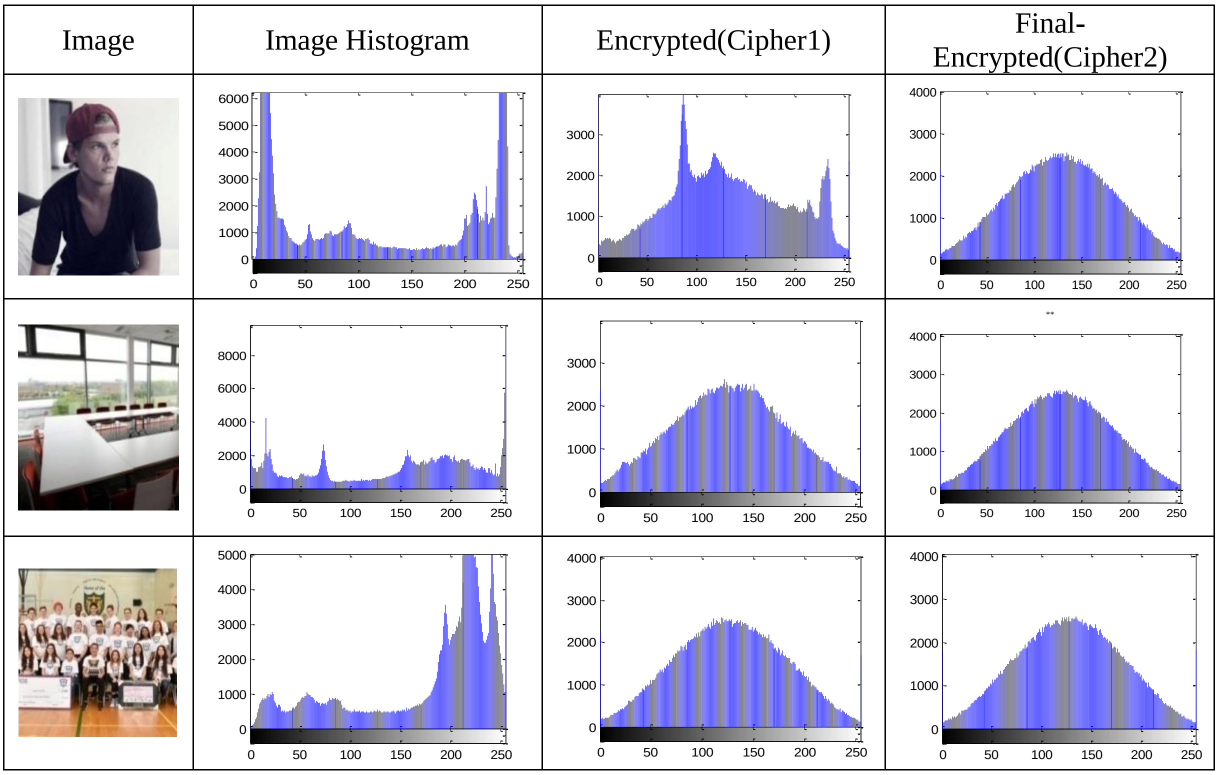


Fig. (5): Source images with their histograms before and after encryption (Cipher1&2)

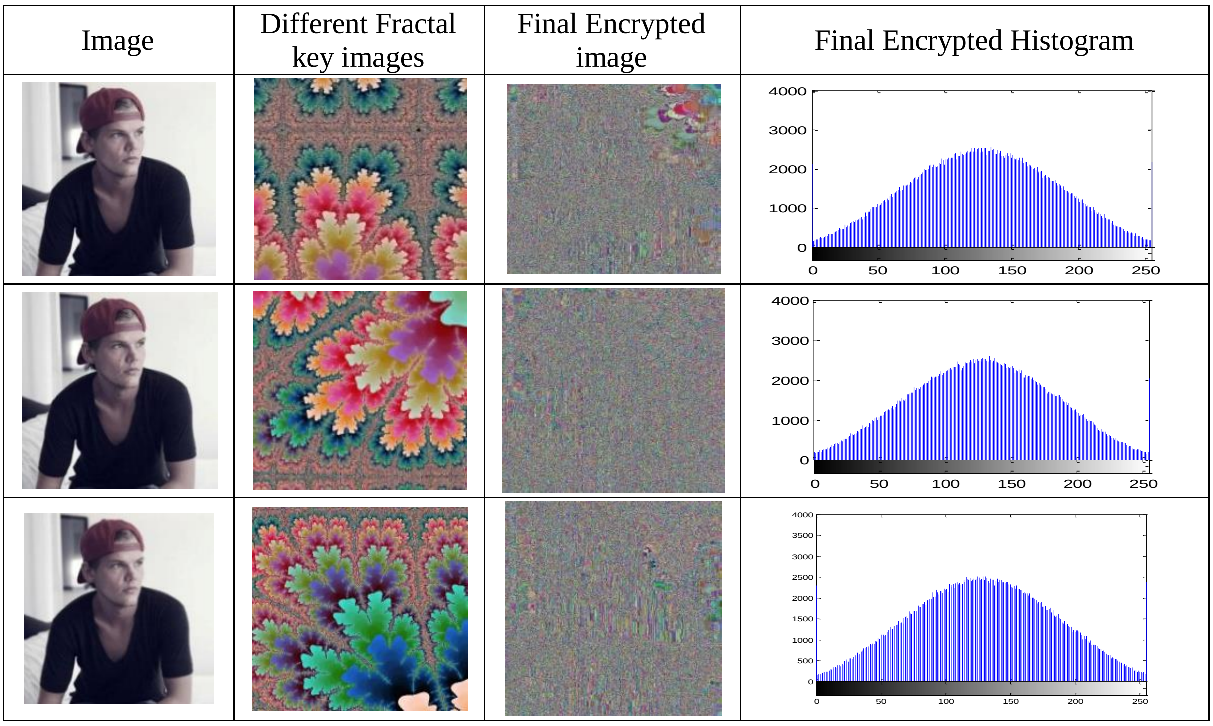


Fig. (6): Source image (Boy) and the encrypted image and histograms using different fractal keys.

Correlation is a simple and useful operation to measure the relationship between two pixels in an image. It is a linear operation that each neighboring pixels in any image have a very close correlation between them. Usually the Correlation coefficient of any original image is close to one as listed in Table (2). To enhances the randomness of the image, correlation coefficient must be reduced for Vertical, Horizontal and diagonal pixels. These correlation coefficients results is obtained using corrcoef() command in the MATLAB application and recorded for the original and encrypted images in Table (2) which shows the degradation in the average correlation coefficients in all encrypted images have changed from 0.99 to be 0.08 and less for all encrypted images.

Table (2): Correlation results for the tested images.

Image	Horizontal Corr.	Vertical Corr.	Diagonal Corr.	Average Corr.
Original-Boy	0.9976	0.9989	0.9968	0.9978
Encrypted	0.1359	0.0736	0.0507	**0.0867**
Original-Room	0.9904	0.9813	0.9722	0.9813
Encrypted	0.0863	0.0622	0.0788	**0.0758**
Original -Group	0.9495	0.9545	0.9133	0.9391
Encrypted	0.1115	0.0480	0.0692	**0.0762**

5. Conclusions

The generation process of the Fractals is time consuming, but on the other hand, it would be a tedious task to avoid the attack and guess the secret key since the fractal geometry is very sensitive to any small change in its parameters. In this paper, a strong and simple hybrid encryption method have been proposed, combining the simple scrambling operation of the XOR technique which broke the correlations of the neighboring pixels of the source image, and the efficiency of the random nature of fractal images that can create thousands of images like Mandelbrot fractal set that is sensitive to a small change in its parameters which was used as a secret key, this key is combined with the managed source image using the private XOR operation. All these methods when used together made the image meaningless and hard to be attacked. Based on the results that are recorded from the experiments we concluded that the performance of the proposed encryption method is very perfect, simple, and have a high space key.

References

[1] A. Uhl., A. Pommer, "***Image and Video Encryption***", Springer Science and Business Media Inc., Advances information and Security, 15, (2005).

[2] D. P. Komal, B. Sonal, "***Image Encryption Using Different Techniques: A Review***", International Journal of Emerging Technology and Advanced Engineering, 1(1), Nov., (2011).

[3] C. H. Chuang, Z. Yen, G. Lin, et al, "***A Virtual Optical Encryption Software System for Image Security***", JCIT, 6(2), (2011).

[4] H. M. Al-Najjar, "***Digital image encryption algorithm based on multi-dimensional chaotic system and pixels location***", Int. J. of Computer Theory and Engineering, 3(4) ,(2012).

[5] S. Kaur, Sh. Angurala, "***Image encryption using chaotic map and prime modulo multiplicative linear congruential generator***", Int. J. of Innovative Research in Computer and Communication Engineering, 3(3), (2015).

[6] A. K. Banthia, N. Tiwari, "***Image encryption using pseudo random number generators***", Int. J. of Computer Applications, 67(20), (2013).

[7] K. C. Murthy, A Mitra, and M. K. Kundu, "***A Study on Partitioned Iterative Function Systems for Image Compression***", Fundamental Informaticae, IOS Press, 34(4), 413, (1998).

[8] A. S. Kamal, R. Ahmed, A. Sherif, and B. Mohamed, "***A fractal-based image encryption system***", IET Image Processing, 8(12), (2014).

[9] M.V. Raju, A. Kumar, and M.V. Madhavi Latha, "***Image Encryption and Decryption Using Scan Pattern***", International Journal of Electronics Electrical and Computational System IJEECS, 5(5), (2016).

[10] G. B. Huntress, "***Encryption using Fractal Key***", United States Patent 6782101, (2004).

[11] ***H. Kashanian, M. Davoudi and H. Khorramfar, "Image Encryption using chaos functions and fractal key***", IJCSNS International Journal of Computer Science and Network Security, 16(10), (2016).

[12] P. S. Addison, "***Fractals and Chaos: An Illustrated Course***", CRC Press, (1997).

[13] http://23programs.blogspot.com/2012/03/c-mandelbrot-set-fractal.html, 8:30 PM, (2/3/2017).

[14] M. Mikhail, Y. Abouelseoud, and G. ElKobrosy "***Two-Phase Image Encryption Scheme Based on FFCT and Fractals***", Hindawi, Publishing Corporation Security and Communication Networks, 1, (2017).

[15] http://en.wikipedia.org/wiki/Mandelbrot set, 9:30 PM, (1/4/2017).

[16] S. Gupta, N. Bansal, "***Image Encryption Techniques using Fractal Geometry: A Comparative Study"***, Journal of Computer Engineering (IOSR-JCE), 16(5), (2014).

[17] http://www.nahee.com/Derbyshire/mandlfaq.html, 12:30 PM, (10/6/2017).

[18] http://www.math.utah.edu/~pa/math/mandelbrot/mandelbrot.html#xyranges, 05:00 PM, (8/3/2017).