IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 16, Issue 5, Ver. I (Sep – Oct. 2014), PP 31-35
www.iosrjournals.org

Image Encryption Techniques using Fractal Geometry: A Comparative Study

Swati Gupta[1], Nishu Bansal[2]
[1](*Department of Computer Science and Engineering, Ajay Kumar Garg Engineering College,Ghaziabad, India)*
[2](*Department of Computer Science and Engineering,Ajay Kumar Garg Engineering College,Ghaziabad, India)*

***Abstract**: Data security has become an important issue in recent times. Confidential data needs to be protected from unauthorized users. Various image encryption techniques have been proposed for the secure transmission of images to protect the authenticity, integrity and confidentiality of images. Fractal images can be used as a strong key for the encryption due to the random and chaotic nature of fractals. The infinite boundaries of fractals provide highly complex structure that leads to confusion and it becomes a tedious task for an unauthenticated user to crack the exact secret fractal key. In this paper, a review of various techniques used for image encryption using fractal geometry has been illustrated. All these techniques have their own advantages and disadvantages in terms of execution time, key generation time and Peak Signal to Noise Ratio.*
***Index terms:** Cryptography, Fractal geometry, Image encryption, PSNR, Security, SSIM.*

I. Introduction

A major issue in transmission of images over a communication channel is to prevent confidential information from being disclosed to illegal and unauthorized users. Image encryption simply means to convert the image into a form that is unreadable and non-understandable. This is done using a special random secret called a key which converts the image to a complex unreadable form. Several techniques and schemes have been proposed over a period of time that guarantees reliable and strong encryption of the image data. Fractal geometry can be used as an efficient technique for image encryption due its inherent nature of random behavior that makes them troublesome to understand. This paper provides a survey on the various fractal based techniques that have been used for image encryption.

Because of the fundamental properties of the images such as high redundancy and bulk data capacity, traditional encryption techniques such as Data Encryption Standard [1] and Advanced Data Standard [2] are not suitable for image data. Firstly, the size of an image is normally greater than that of the text due to which traditional techniques require more time to decrypt the encrypted images. Secondly, the decrypted text data must be same as the original text or plaintext. But due to the characteristics of human perception, decrypted image having a small amount of distortion is acceptable. Hence there is a requirement of separate techniques for image encryption.

Fractal geometry [3] is the geometry of irregular shapes which are characterized by infinite detail, infinite length, and the absence of smoothness. 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. Unlike Euclidean geometry, fractal curves possess non-integer dimensions. The fractal curves have dimensionality between 1 and 2 [4]. The Mandelbrot set [5] is the best known fractal; it is defined as the set of all points C_0 in the complex plane, such that the infinite sequence $C_o, C_1, \ldots, C_n, \ldots$ remains bounded, where:

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

Most of the techniques use Mandelbrot fractal curve as the encryption key. A typical Mandelbrot set is defined by [6]:

$$f : z_{n+1} \leftarrow z_n^{\,2} + c \quad (2)$$

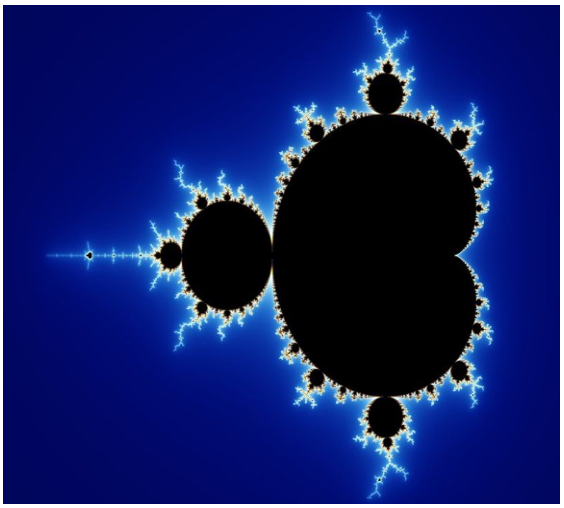

Fig. 1. Mandelbrot fractal set [7].

II. Related Work

Fractals were first used for text encryption by Jhansi et. al [8] where self symmetric property of Sierpinsky fractal was used. The fractal curve was not used as a fractal key, only its geometry was used to encrypt the plaintext with the help of secret key (text). Sierpinsky triangle is a fractal that is generated by using a triangular initiator and connecting the mid-points of the three edges of the triangle to form four smaller triangles. On iterating this process on the smaller triangles a mathematical set which is known as Sierpinsky triangle is obtained [9].

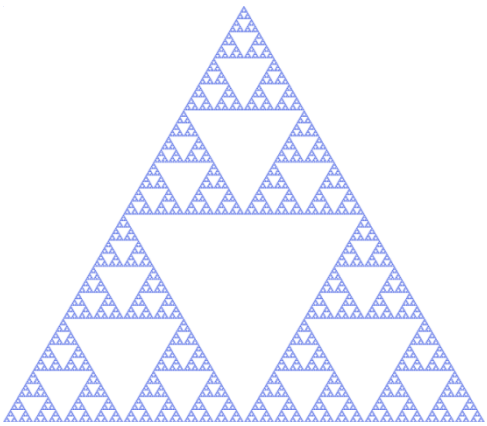

Fig. 2. Sierpinski Triangle [9].

The secret key bit was placed at the center while the plain text bit is placed in the surrounding triangles. Encryption algorithm used left-right circular shift operation and XOR-XNOR operations. The proposed algorithm is resistant to ciphertext only attacks, known plaintext attack and chosen plaintext/ciphertext attack and is on par with DES. This technique was used only for text data, hence there is a requirement for a robust technique for encryption of images.

Using fractals for image encryption was first proposed in [10] where a selected square fractal key matrix was used as a fractal key. Matrix multiplication was used as the operation for encryption. The fractal encryption key is computed iteratively, changing fractal parameters as necessary, to obtain a full rank [11] fractal key matrix of size (NxN). The image data to be encrypted is buffered to form a two-dimensional matrix of size (JxN) or (NxJ) where J any be any positive integer including N. For a buffered data matrix of size (JxN), encryption uses matrix multiplication using the equation:

$$E = P * K \tag{3}$$

where:
P = the buffered data matrix (JxN);
K = the fractal key matrix (NxN);
* indicates matrix multiplication; and
E = the encrypted data matrix (NxJ).

After encryption, the encrypted matrix is sent to the receiver along with the fractal key parameters so that the same matrix can be generated at the receiver side and can be used as the decryption key. The transmitted fractal key parameters must contain all the information necessary for the receiver to generate the fractal key, including but not limited to the fractal choice, any fractal initialization values, the fractal key matrix size (N), and the encrypted data matrix size (NxJ) or (JxN). The receiver performs the decryption using the available information using the equation:

$$P = E * K^{-1} \tag{4}$$

where:

K^{-1}=the inverse fractal key matrix.

The proposed way of multiplication of the image matrix and the key matrix (the inverse key matrix) for encryption (decryption) is slow when it comes to big images. Also, it may require reformatting and post-processing at the receiver side. On applying the algorithm on a selected plainimage using a Mandelbrot curve with certain parameters, we get a low PSNR value of 11.7241 for Red layer, 11.0069 for Green layer and 10.6269 for Blue layer.

Another image encryption technique based on the modulo operation with fractal keys was proposed in [12]. The Mandelbrot set was generated using the bound set of S obtained from the recursive formula:

$$\forall \bar{x} \in C$$

$$S_{n+1}(\bar{x}) = \left[S_n(\bar{x}) \right]^a + b \times S_o(\bar{x})$$

$$a \in R, b \in C \tag{5}$$

For each pixel of this fractal image, a grid D' is constructed with spacing δ for all the three layers R, G, and B independently. For each layer, encryption is done using the equation:

$$e'_{ij} = (e_{ij} + d'_{ij}) \bmod 256 \tag{6}$$

The decryption process includes the key generation at the receiver side and then using the equation for each layer:

$$[\mathrm{e_{ij}}] = ([\mathrm{e'_{ij}}] - [\mathrm{d'_{ij}}]) \bmod 256 \tag{7}$$

where mod denotes the modulus operation.

The strength of the algorithm is determined by conducting the PSNR [13] test. The results show that the algorithm is strong enough to stand up to even the slightest changes in the key parameters and hence it is invulnerable to brute force attack [14] because the fractal key generation is time consuming. The value of the spacing δ must be small. The smaller the value of δ, the more secure encryption we achieve. This value cannot be extremely small nor can it be extremely large.

The PSNR values as computed between the original image and the encrypted image by using a Mandelbrot curve created using specific parameters are 12.8619 for Red layer, 13.9165 for Green layer and 11.0339 for Blue layer.

A technique based on matrix multiplication was proposed thereafter that involves compression of the image using fractal codes and then multiplying it with the selected fractal image matrix [15]. Both the matrices are first permuted which enhances the randomness in both the images.

$$C = P(A) \tag{8}$$

$$D = P(B) \tag{9}$$

$$E = C \times D \tag{10}$$

where P denotes the permutation operation and X denotes the multiplication operation.

The encrypted image E is sent to the receiver and the decryption is performed using the equations:

$$E \times D^{-1} = C \times D \times D^{-1} \tag{11}$$

$$C = E \times D^{-1} \tag{12}$$

$$A = P(C) \tag{13}$$

where:

A = the plainimage

C = plainimage after permutation

B = chosen fractal image

D = fractal image after permutation

E = decrypted image

Since fractal image has a very small memory footprint, memory requirement is quite low. Moreover, a slight difference in fractal parameters changes the fractal key to a great extent which makes the algorithm highly key sensitive. The key generation process is time consuming making it invulnerable to brute force attack. The algorithm results in a low PSNR value of 9.0470 for Red layer, 8.55 for Green layer and 7.58 for Blue layer.

Mandelbrot set in a combination with Hilbert transformation has been used to enhance the randomness of the secret key in [16]. The Hilbert curve is a way of mapping the multidimensional space into a one-dimensional space. The Mandelbrot curve matrix is considered as the three layer matrix with RGB components. Each pixel *i(x,y)* is mapped to a one-dimensional coordinate. An integer *r* is calculated that is the interval distance between one point to another while travelling the curve, to calculate the value of the pixel again.
Each pixel in the image to be encrypted is labeled *O(x,y)*. *K(x,y)* denotes the key and *T(x,y)* denotes the encrypted image. The encryption is done following the equation:

$$T(x,y) = \left(O(x,y) + K(x,y)\right) mod\ 256 \tag{14}$$

The decryption is done following the equation:

$$T'(x,y) = \left(O(x,y) - K(x,y)\right) mod\ 256 \tag{15}$$

where mod denotes the modulus operation.

The encrypted image and the decrypted image are compared using the PSNR test which shows that the encrypted and the decrypted images are the same.

Mandelbrot fractal set in combination with Julia fractal set was used for a new key exchange protocol in [17]. The algorithm is based on the concepts of traditional Diffie-Hellman key exchange protocol [18]. The secret key is exchanged between the two users without any prior communication between them over an unsecure channel. The Mandelbrot fractal set can be generated using the following equation recursively:

$$Z_n = Z_{n-1}^{\ 2} + c; Z_0 = 0; c, Z_{n-1} \in C; n \in Z \tag{16}$$

where C is the set of points in the complex plane. Similarly, the Julia fractal set is a set of points on a complex plane defined recursively as:

$$Z_n = Z_{n-1}^{\ 2} + c; c, Z_n \in C; n \in Z \tag{17}$$

The proposed protocol uses *c* as a global value which is known to the public, e and n as private value for the sender, and k and d as private values for the receiver. The private values and the global value c are used as the inputs to the Mandelbrot function, and result in two public keys, $Z_n e$ for the sender and $Z_k d$ for the receiver. With the private information and the other party's public key as inputs to the Julia function, the same secret key can be generated as both sides.

$$(Z_n e)_k d = (Z_k d)_n e \tag{18}$$

The computed results show that the proposed algorithm is more efficient than the Diffie-Hellman key exchange algorithm both in terms of key size and execution time.

TABLE I. Comparision table

Author	Year	Technique	Advantages	Disadvantages
P. Jhansi Rani, Durga Bhawani	2011	Symmetric encryption using Sierpinski triangle	Resistant to ciphertext only attack, known plaintext attack and chosen plaintext-ciphertext attack	Used for text encryption, not for images
G. B. Huntress	2004	Matrix Multiplication	Secure than traditional schemes	Slow, Require reformatting and post-processing after decryption
Valerij Rozouvan	2009	Modulo Operation	Invulnerable to brute force attack, small memory footprint required	Slower to encrypt single images
A. J. J. Lock, C. H. Loh, S. H. Juhari, A. Samsudin	2010	Multiplication of permuted matrices	Image compression reduces redundancy, memory requirement is low	Key generation is time consuming
Y. Y. Sun, R. Q. Kong, X. Y. Wang, L. C. Bi	2010	Use of Hilbert Curves with modulo operation	High key sensitivity	Time consuming
M. A. Alia, A. B. Samsudin	2007	Key exchange protocol	Small key size, less execution time	-

III. Conclusion

Various techniques involving fractal geometry for image encryption are studied. All these techniques have their own advantages and disadvantages. The use of fractals enhances the perplexity of the encrypted image but at the same time it may prove to be complicated to the user itself. Fractal generation process is time consuming which is a disadvantage to the user as the whole encryption process will become time consuming, but on the other hand it is advantageous because cryptanalytic attacks such as brute force attack can be avoided due to the same. This is so because it would be a very tedious task for the attacker to guess the secret key since the Mandelbrot geometry is highly sensitive to a minute change in its parameters.

The techniques can be classified based on PSNR values that define the Peak Signal to Noise Ratio between the original plainimage and the encrypted image. The Mean Square Error (MSE) must be maximum between the original image and the encrypted image and hence the PSNR must be low because PSNR is inversely dependent on the MSE.

Another metric to judge the image encryption techniques can be Structural Similarity (SSIM) Index [19]. SSIM is a full reference metric to measure the similarity between two images based on image quality. It considers the idea that the image pixels are strongly inter-dependent on each other when they are spatially close. SSIM does not estimate perceived errors; rather it is based on perceived change in structural information. While PSNR depends on the Mean Square Error only, SSIM takes into account variance and covariance between the two windows that are compared. Hence, SSIM proves to be consistent with human eye perception and provides more accurate results.

References

[1] National Institute of Standards and Technology, Data Encryption Standard (DES), Technical Report, Federal Information Processing Standards Publication 46-3, 1999.
[2] National Institute of Standards and Technology Advanced Encryption Standard (AES), Technical Report, Federal Information Processing Standards Publication 197, 2001.
[3] D. Hearn, M. P. Baker, Computer Graphics C Version (Second Edition, Prentice Hall, pp. 362-387).
[4] W. S. Maki, Simple Geometric Fractals, Behavior Research Methods, Instruments and Computers, Psychonomic Society, Inc. 23(2), pp. 160-165, 1991.
[5] V. Rozouvan, Symmetry of the Modified Mandelbrot Set, Pi in the Sky, Pacific Institute for Mathematical Sciences (PIMS), Issue 9, December 2005.
[6] http://en.wikipedia.org/wiki/Mandelbrot_set
[7] http://en.wikipedia.org/wiki/File:Mandel_zoom_00_mandelbrot_set.jpg
[8] P. Jhansi Rani, Durga Bhavani, Symmetric Encryption using Sierpinsky Fractal Geometry, International Conference on Information Processing, ICIP 2011 Proceedings, Springer-Verlag Berlin Heidelberg, pp. 240-245,2011.
[9] http://en.wikipedia.org/wiki/Sierpinski_triangle#mediaviewer/File:Sierpinski_triangle.svg
[10] G. B. Huntress, Encryption using Fractal Key, United States Patent 6782101, 2004.
[11] http://www.mu.ac.in/myweb_test/syllFybscit/APM.pdf
[12] V. Rozouvan, Modulo Image Encryption with Fractal Keys, Optics and Lasers in Engineering, Vol. 47, issue 1, pp. 1-6, 2009.
[13] Q. Huynh-Thu, M. Ghanbari, Scope of Validity of PSNR in Image/Video Quality Assessment, Electronics Letters 44 (13), pp. 800-801, 2008.
[14] L. R. Knudsen, M. J. B. Robshaw, The Block Cipher Companion, Information Security and Cryptography, pp. 95-108, 2011.
[15] A. J. J. Lock, C. H. Loh, S. H. Juhari, A. Samsudin, Compression-Encryption Based on Fractal Geometric, Second International Conference on Computer Research and Development, IEEE Computer Society, 2010.
[16] Y. Y. Sun, R. Q. Kong, X. Y. Wang, L. C. Bi, An Image Encryption Algorithm using Mandelbrot Set, International Workshop on Chaos Fractal Theory and its Applications, IEEE Computer Society, 2010.
[17] M. A. Alia, A. B. Samsudin, New Key Exchange Protocol Based on Mandelbrot and Julia Fractal Sets, IJCSNS International Journal of Computer Science and Network Security, Vol. 7, No. 2, February 2007.
[18] W. Stallings, Cryptography and Network Security Principles and Practices (Pearson Education, 3rd edition).
[19] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, Image Quality Assessment: From Error Visibility to Structural Similarity, IEEE Transactions on Image Processing, Vol. 13, No. 4, April 2004.

Biography

Swati Gupta received her B.Tech (Computer Science and Engineering) degree from Mody Institute of Technology and Science, Lakshmangarh (Rajasthan) in 2012. She is currently pursuing her M. Tech (Computer Science and Engineering) from Ajay Kumar Garg Engineering College, Ghaziabad (Uttar Pradesh). Her areas of interest include Computer Graphics, Software Engineering and Discrete Mathematics.

Nishu Bansal got her B.Tech degree from UP Technical University in 2005.She has done her M.Tech from Guru Gobind Singh Indraprastha University, New Delhi. She has around 9 years of teaching experience. She is an Assistant Professor in the Department of Computer Science and Engineering of AKGEC, Ghaziabad affiliated to Uttar Pradesh Technical University. Her areas of interest include programming languages, computer architecture, Search Engine, Swarm Intelligence and Adhoc Networking.