International Journal of Current Advanced Research
ISSN: O: 2319-6475, ISSN: P: 2319-6505, Impact Factor: 6.614
Available Online at www.journalijcar.org
Volume 7; Issue 5(L); May 2018; Page No. 13006-13010
DOI: http://dx.doi.org/10.24327/ijcar.2018.13010.2308

Research Article

A COMPREHENSIVE RESEARCH ON IMAGE CRYPTOGRAPHY USING MBST FOR EFFICIENT ENCRYPTION

Manoshi Mistry[1] and Nidhi[2]

[1]DPGITM, Maharshi Dayanand University, Haryana, India
[2]Department of Computer Science and Engineering of DPGITM, Maharashi Dayanand University, Haryana, India

A R T I C L E I N F O

Article History:

Received 24th February, 2018
Received in revised form 19th March, 2018 Accepted 16th April, 2018
Published online 28th May, 2018

Key words:

Cryptography, Encryption, Decryption, MBST, AES, MAES

A B S T R A C T

This paper focussed on the security management of an image over a network which includes user authentication, accuracy, integrity and security of the sent images. Additionally, multimedia files consist of image, audio or video as data require more attention during encryption and decryption. The Proposed layout of image encryption/decryption uses a suitable user-defined key which is designed to obtain the research objective. In this paper, a cryptographic transformation based on matrix manipulations for image encryption is introduced. The paper describes the encryption algorithmusing "Matrix Block Symmetric Transformation (MBST)", discusses the simulation results and compares with results obtained from Advanced Encryption Standard (AES) and Modified Advanced Encryption Standard (MAES).

INTRODUCTION

In this modern era, privacy of the multimedia data becomes more crucial and necessary [1]. Application of text, image, audio and video encryption is necessary in several fields which include internet communications, medical imaging, telecommunication, multimedia systems, and military assistance [2]. So, security of any kind of data files whether text, image, audio, video etc. became an important parameter during transmission and storage with the advancement in the technology. This led to a growing demand of advance cryptography. This paper mainly focussed onimage encryptionfor image data hiding from unauthorized access. Encryption algorithms for image hiding range from very simple to most complex spatial as well as frequency domain techniques. Secure image-data communication means strong encryption technique at the transmission and a safe decryption at the receiving end. As, an image encryption is somehow shows similarity as well as a slight difference from general text data encryption so, this researchpaper approached to a new way of image encryption merging certain common and contrastingfeatures of traditional cryptosystem techniques.

The type of image data could be .jpg or .bmpin the entire methodology of image encryption and decryption process wherein the proposed method has shown the result with the help of histogram of the encrypted and decrypted image.

Corresponding author:* **Manoshi Mistry
DPGITM, Maharshi Dayanand University, Haryana, India

Moreover, an image usually has a bigger size than that of text due to which a cryptosystem takes a longer time in direct encryption of the image data. Also, large size image further takes longer encryption time before its transmission, thus, gradual delays in the successive transmission of encrypted images could be expected. As a result, an efficient encryption algorithms are needed to minimize the latency. In multimedia encryption applications, an encryption algorithm must achieve a high throughput along with an adequate security. Traditional block ciphers like DES (Data Encryption Standard) [3] and AES (Advanced Encryption Standard) [4] are not recommended especially for a larger image size [5]. Two types of cryptography viz. Symmetric key cryptography and public keycryptography are basics for an information encryption in which Public-key algorithms considerably run slower as compare to Symmetric-keyalgorithms [6]. To resolve the traditional problem of data transmission or communication over an insecure channel, Symmetric-keycryptography is still extensively preferred [7]. In the field of high-speed networking, high throughput encryption and decryption are getting increasingly demandable [8,9]. High-speed secure multimedia data communication requires fast encryption [10]. In this paper a fast symmetric key encryption procedure, MatrixBlock Symmetric Transformation (MBST) based onmatrix manipulation is presented. The encryption schemepresented here is a block cipher with a block size of 128 bitsand key size of 128 bits.

Therefore, the mathematical parameters viz.the information entropy and the elapsed time have been used in statistical analysis.

Please note that as per a human perception, a minor distorted decrypted image data is acceptable unlike text-data decryption in which decrypted text (whether letters or numerals or alpha-numeric) must be identical to its original form.

Cryptography

Overview

This entire paper is categorised into four sections that includes:

- Section I: Basic introduction to image encryption and problem formulation
- Section-II: Detailed description of proposed MBST algorithm for encryption
- Section-III: Experimentand Simulation result
- Section-IV: Conclusion and future enhancement.

Encryption & Decryption Architecture

The proposed architecture of the image encryption algorithm (as shown in Fig.1) is similar to that of the basic cryptosystem which is further defined in comparison to the modern encryption algorithms (as shown in Fig. 2).

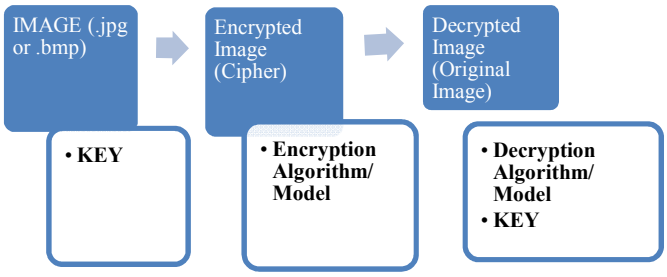

Fig 1 Cryptography Architecture

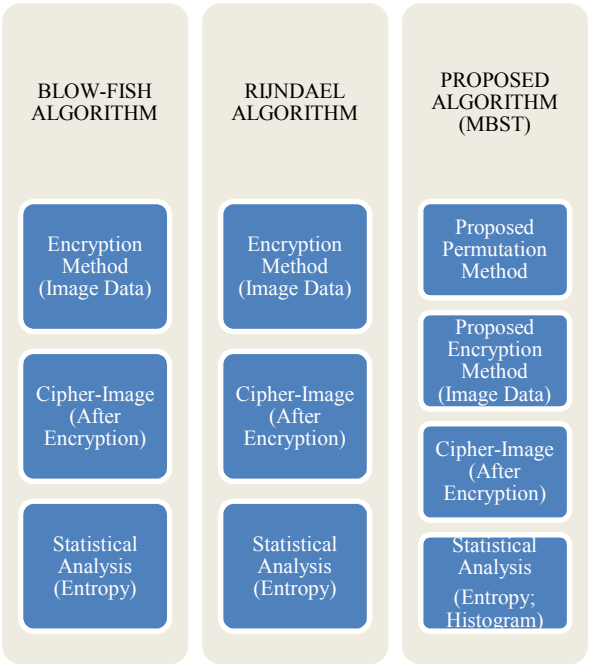

Fig 2 Block Diagram (Mbst Vs Other Algorithms)

Matrix Block Symmetric Transformation (MBST)

Matrix block based manipulated encryption algorithm includes two types of keys:

- Secret key and
- Sub keys.

Three functionalcomponents for generating SMBT keys are as follows:

- Matrix Generation
- Key Arrangement
- Replacement and Diffusion

Matrix Generation

Requirements

An Encryption Matrix(E)
Size (E) = 16 x 256 bytes
Number range (E) = 0-255 (incl. 0 and 255)
Rows (E) = 16
Columns (E) = 256

A Secret Key (K)
Rows (E) = Decimal values of characters of chosen key from K
Shuffling of the matrix columns is done by a table look-up procedure.
The order of numbers in E is based on the characters of K. For instance,
The first column in the i^{th}row of thematrix is filled as: E[i][1] = Integer value of K[i]
Similarly, the entire matrix E is generated as per the Pseudo-code 1.

```
For i = 1 to 16//Number of rows
For j = 0 to 256            //Number of columns
E1(i,j) = int(K(i)) + (j-1);
If E1(i,j) > 255
{
E1(i,j) = E1(i,j) – 256;
}
EndFor   //Ending columns
EndFor   //Ending rows
```

Pseudo-Code 1: Matrix Generation

Significance

- During 'Key Arrangement' procedure, generation of the sub keys for diffusion function.
- During 'Replacement and Diffusion' procedure, substitution of a value received from a selected row of matrix E to the input data byte.

Key Arrangement

Requirements

16 pairs of sub keys:

Generation of sub-key matrices, K1 and K2 from basic encryption matrix, E
Such that,
Size (K1) = 16 x 16 bytes
Size (K2) = 16 x 16 bytes.
Number of diffusion rounds =16 (a typical block cipher)

Complex key arrangement:

To avoid the unauthorised entity to derive the sub-keys.

Diffusion in sub-keys:

A small change in the secret key must clearly diffuse into the sub-keys.

For e.g., one-bit change in secret key must reflect several bits change in the sub-keys.

Procedural steps for Pseudo code 2

Secret key, K, is firstly fed into a byte-level transposing operation (T) to get KA1 that would interchange the position of the MS byte and the LS byte.

For an instance,
Bytes in array(K) = {K0, K1, K2, K3, …., K14, K15}
After Transposition,
Bytes in array (KA1) = Transposed (K) = {K15, K14, …., K5, K4, K3, K2, K1, K0}

An XOR operation between KA1 and K with an output in KA2.

This operation can cause up to 2-bits change in KA2 when 1-bit is changed in secret key K.

An XOR operation between Left half of KA2 and right half of KA2 with an output in KA3

Individually, transposing left half of KA1 and right half of KA2 then XOR operation between the obtained transposed elements with an output inKA4.

A concatenation of KA3 and KA4 with an output in KA5.
This operation would lead up to 4-bits change in KA5 when 1-bit change in secret key, K.

Computation of the sum of integer values of bytes in KA5 with an output in L, such that,
KSE1 = L % 23 and
KSE2 = L % 15.
This operation would lead up to 4-counts change in KSE1 and KSE2 when 1-bit change in secret key, K.
Thus, (KSE1 + KSE2) can have up to 8 counts change.

Two matrices KS1 and KS2 of size 16 x 16 bytes are generated from the base matrix, E, such that
KS1[row][column]=E[row][KSE1+KSE2+column]
KS2[row][column]=E[row][KS1[row][column]]
Where,
Columns of KS1 matrix are selected from the base matrix, E based on KSE1 and KSE2 values and
Columns of KS2 matrix are selected from the base matrix, E based on the element values of the columns of KS1 matrix.
This operation would lead up to 8-counts change in KS1 when 1-bit change in secret key, K.

Rotating vertically down i^{th} column of matrix KS1
where, number of times = ((int(K[i]) % 12) + KSE1).

Rotating vertically down i^{th} column of matrix KS2
Where, number of times = ((int(K[i]) % 10) + KSE2).

The vertical rotation would shuffle the elements of sub-key matrices thereby providing more changes in the sub-key values while one-bit change is applied on the original secret key, K.

```
KA1 = Transpose (K);
KA2 = XOR (KA1, K);
KA1 = Shift Left (K, 8 characters);
KA2 = Shift Right (K, 8 characters);
KA3= XOR (KA1, KA2);
KA1 = Transpose (KA1);
KA2 = Transpose (KA2);
KA4 = XOR (KA1, KA2);
KA5 = Concatenate (KA3 + KA4);
L = int (SUM (Elements of KA5));
KSE1 = L % 23;
KSE2 = L % 15;
For r ← 0 to 15
{
For i ← 0 to 15
{
KS1[i][r] = E[i] [(KSE1 + KSE2 + r)];
KS2[i][r] = E[i] [(int) KS1[i][r]];
KS1[i][r] = E[i] [(int) KS2[i][r]];
Q[i] = (int)K[i] % 12;
}
}
EndFor
EndFor
For i ← 0 to 15
{Circular shift down i^th column of KS1 & KS2;
number of times = KSE1+ Q[i];}
EndFor
```

Pseudo-Code 2: Key Arrangement

Replacement and Diffusion Iteration

Significance

Transformation of the plain-image into cipher-image in 16 bytes blocks.

Procedural steps for Pseudo code 3

In the Cipher, 16 iterative units of 'Replacement and Diffusion' is shown in a data block, P, that consists of 16 bytes.

At the input of iteration round, P(0) to P(15) is put in.

Different value of data byte, P(i), is considered in which the decimal value of P(i) isused to read the value E(i,j) as the column number, j, of the j^{th} row of base matrix, E. This value is taken as the substitute for P(i).
For instance,
for the byte P(0) in a block,
i = 0 and j = decimal value of P(0) use to determine the value E(0,j)
Substitute this E(0,j) in C(0) for P(0).
Similarly, all the 16 bytes are substituted in data block by taking the value from the selected row and column of the base matrix, E that depends on the position of the block data and the data value.

As per the Pseudo code 3, the diffusion of data is provided by the following operations:

- key based XOR,
- data based XOR,
- transposition (T) and
- data based rotation (>>>) operations.

Number of iterations would be selected between 1 and 16 and the output of last iteration round is taken as the Cipher-text block.

```
For a ← 0 to 15
{b = P(a);
C(a) = EM[a][b];}
EndFor
C = EXOR (C, KS1);
C = Transpose (C);
C1 = left half bytes(C);
C2 = right half bytes (C);
C2 = EXOR (C1, C2);
C1 = Transpose (C1);
C1 = EXOR (C1, C2);
C3 = C1 || C2;
C3 = Transpose (C3);
C3 = EXOR (C3, KS2);
C4 = Right half bytes(C3);
C5 = left half bytes (C3);
C6 = byte (SUM(C5));
C6 = MODE (C6, 6);
```

```
Rotate right C4, Number of times = C6;
C7 = byte (SUM(C4));
C7= MODE (C7, 6);
Rotate right C5, Number of times = C7;
C=C4 || C5;
P=C;
```

Pseudo-Code 3: Replacement and Diffusion

Simulation Results

The encryption algorithm has been tested with different images of various sizes and the results are compared with AES and MAES.

The simulation tests have been conducted using MatlabR2018a in an Intel(R) Core(TM) i7-6500U 2.50GHz processor with Windows 10 operating system.

Measurement of elapsed time, encryption speed and information Entropy tests have been made.
The similar key is used to decrypt the encrypted image.

Histograms analysis of encrypted and decrypted images

The histograms of encrypted and decrypted images have the distribution of the pixels in grey values and significantly different from the original image. Fig. 3, 4 and 5 shows the cipher images obtained using MBST key with their respective histograms.

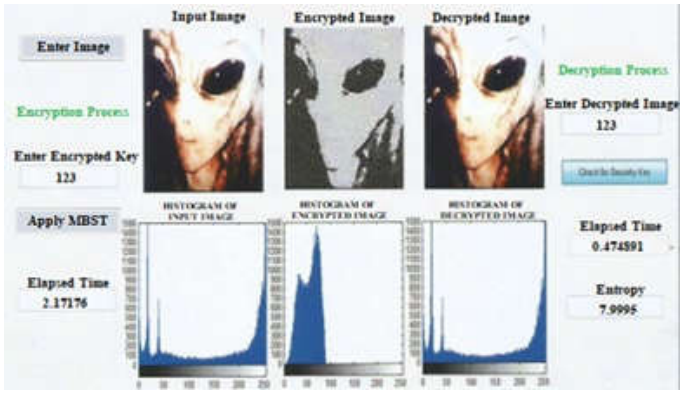

Fig 3 Histogram of Encrypted and Decrypted Image 1

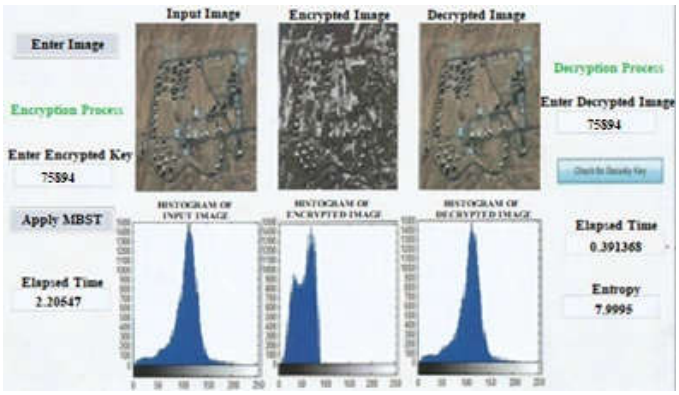

Fig 4 Histogram of Encrypted and Decrypted Image 2

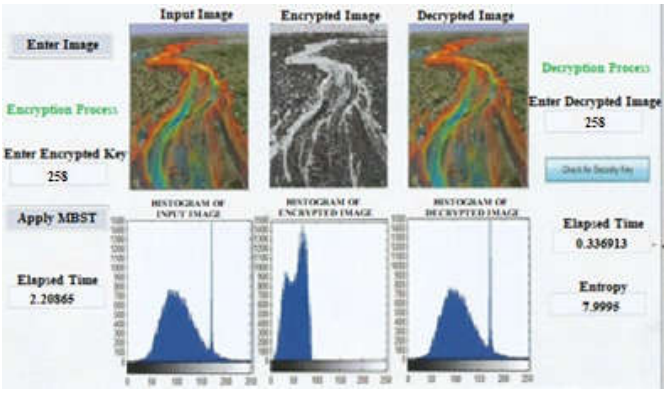

Fig 5 Histogram of Encrypted and Decrypted Image 3

Elapsed Time and Encryption Speed

Image processing time of the entire operation is referred as 'Elapsed Time'. In the proposed technique, the encryption and decryption time of the image using MBST key is termed here as elapsed time. Since the encryption key needs to set and later the reverse process of the encryption is needed to perform. Therefore, the value of encryption and decryption elapse time vary with respect to the set key value.

The proposed algorithm is implemented to measure the encryption speed. Two images of size 256 x 256 each and one image of size 512 x 512 were chosen for the observations and measurements.

Table 1 shows the encryption speed observed using these images for different rounds in SMBT.

The average of encryption time and decryption time is used to computethe speed in bytes / second.

Table 1 Elapsed Time and Encryption Speed Data

Image Size (in Pixel)	Image Size on Disk	Encryption Time (in ms)	Decryption Time (in ms)	Encryption Speed (in bytes/second)
Image 1 (256 x 256)	39.6 KB	2.17176	0.474891	29924.6104
Image 2 (256 x 256)	40.1 KB	2.20547	0.391368	30883.7132
Image 3 (512 x 512)	66.7 KB	2.20865	0.336913	52404.91

Information Entropy

The mathematical theory of 'Information Entropy' is traditionally, founded by C.E. Shannon in 1949 [11] for the study of data storage and communication. Today, Information theory is used for:

- Cryptography
- Communication Systems
- Compression of data
- Error – correction and in many other subjects or topics

Information Entropy H(s) of a source 's' can be computed using:

$H(s) = \sum_{i=0}^{2^N-1} P(s_i) \cdot \log_2 \frac{1}{P(s_i)}$ (bits)

Where,
s_i = Symbol,
$P(s_i)$= Probability of symbol s_i
For instance, a source emits symbols with equal probability of 28 counts such that $s_i = \{ s_1, s_2, \ldots\ldots\ldots\ldots, s_{28} \}$.

After computation, for a random source, H(s) = 8.

This concludes a smaller entropy value than that of ideal figure for a practical information source that randomly generates message information. On the contrary, the encrypted message must generate an ideal value of 8. This is due to the fact that a cipher emitted symbol with an entropy less than 8 could cause an extent of predictability thus, it could be a security threat.
The accuracy of proposed algorithm using MBST key is compared to that of AES and MAES with the help of information entropy evaluation parameters as shown in figure 6.

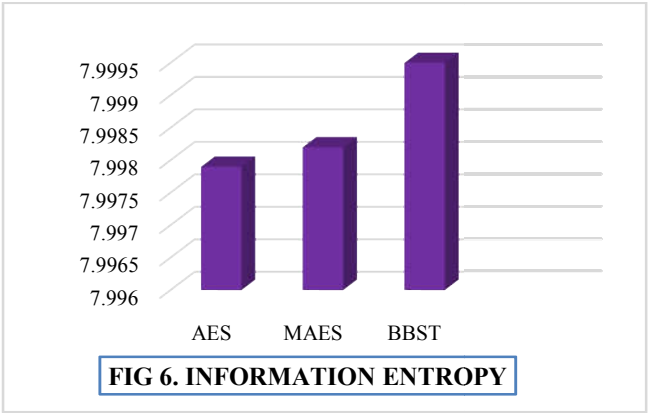

FIG 6. INFORMATION ENTROPY

CONCLUSION AND FUTURE SCOPE

This paper focused on comprehensive research of the image cryptography using MBST approach to prevent image data from hackers/cryptanalysts. The strength of the secret key decides the strength of a cryptographic algorithm. Thus, the proposed algorithm successfully achieved its objective by verifying results using parameters like elapsed time, encryption speed and the most importantly with the help of 'Information Entropy'. Additionally, this paper compared the information entropy result of the proposed image encryption algorithm with one of the popularly known encryption algorithms viz. AES and MAES. Moreover, one of the advance key functions viz. MBST function is mentioned to be used as key in image encryption.

There is an open space for modifications in the proposed algorithm. It could bemodified to embedit withother types of data such as for audio, video or any other form of multimedia data in near future.

References

1. Ju Young O.H et-al, "A Selective Encryption Algorithm based on AES for Medical Information," Health Informatic Research, Vol 16, No. 1, March 2010, pp. 22–29.
2. M.Zeghid et-al, "A Modified AES based Algorithm for Image Encryption," *World Academy of Science, Engineering and Technology*, 27 2007, pp 206 – 210.
3. Data Encryption Standard: http://csrc.nist.gov/publications/fips/fips 46-3/fips- 46-3.pdf
4. Advanced Encryption Standard: http://csrc.nist.gov/publications/fips/fips197/ fips-197.pdf
5. Xiaogang Jia *et-al*, "Image Encryption using IKEDA Map," International Conference on Intelligent Computing and Cognitive Informatics, IEEE Computer Socoety, pp. 455-458 , (2010)
6. Jose J. Amador, Robert W. Green, "Symmetric-Key Block Ciphers for Image and Text Cryptography," *International Journal of Imaging System Technology*, Vol. 15 – pp. 178-188, (2005).
7. Dragos Trinca, "Sequential and Parallel Cascaded Convolution Encryption with Local Propagation: Toward Future Directions in Cryptography," Proceedings of The third International Conference on Information Technology-New Generations. (ITNG'06), 0-7695-2497- 4 / 2006, IEEE Computer Society, (2006).
8. Adam J. Elbirt, Christof Paar "An Instruction- Level Distributed Processor for Symmetric-Key Cryptography," IEEE Transactions on Parallel and distributed Systems, Vol. 16, No. 5, May, (2005).
9. Paul A.J., Varghese Paul, P. Mythili "Fast Symmetric Cryptography using Key and Data based Masking Operations," *International Journal of Computational Intelligence-Research and Applications*. Vol. 3, No. 1, January-June 2009, pp 5–10.
10. Krishnamoorthy G.N, V. Ramaswamy, "Encryption Quality Analysis and Security Evaluation of CAST-128 Algorithm and its Modified Version using Digital Images," *International Journal of Network Security & its Applications*, Vol.1, No.1, April 2009, pp 28-33.
11. Shannon, C. E. (1949) "Communication Theory of secrecy systems", *Bell system technical journal*, 28(4), 656-715.

How to cite this article:

Manoshi Mistry and Nidhi (2018) 'A Comprehensive Research on Image Cryptography Using Mbst For Efficient Encryption', *International Journal of Current Advanced Research,* 07(5), pp. 13006-13010.
DOI: http://dx.doi.org/10.24327/ijcar.2018.13010.2308

****