ISSN:0975-9646

Deep Desai et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 3 (4) , 2012,4809-4811

Chaos-Based System for Image Encryption

Deep Desai[1], Appoorv Prasad[2], Jackson Crasto[3]

Computer Department, Mumbai University
St.Francis Institute Of Technology,

Abstract— **We propose an algorithm that will allow encryption of image data. Traditional symmetric ciphers such as Advanced Encryption Standard (AES) are designed with good confusion and diffusion properties. These two properties can also be found in chaotic systems which have desirable properties.**
Our algorithm uses chaotic maps. Chaotic maps demonstrate great potential for information security, especially image encryption. Chaotic output signals, which present random statistical properties, are used for both confusion and diffusion operations in a cryptosystem.
The system uses combinational permutation techniques that divides the image into blocks, and then shuffles their positions before passing them to the bit manipulation stage. At bit manipulation stage the system will modify individual pixel values using an encryption key.
A chaotic map will be used to perform permutation i.e. diffusion on each of the pixel value. As chaotic map is applied all the pixel position will be scattered. During bit level manipulation the bits for each pixel are permuted to achieve diffusion at bit level.

Keywords— **Chaotic map, Confusion, Diffusion.**

I. INTRODUCTION

Internet communication has become an integral part of the infrastructure of today's world. The information communicated using internet connectivity comes in numerous forms and is used in many applications. In a large number of these applications it is desired that the communications be done in secret. Data secrecy had become an important issue.

Encryption provides an obvious approach to information security and encryption programs are readily available. Encryption algorithms available for textual data are highly efficient. But sometime the information is available in form of image. In such cases we need a specialized algorithm that is highly optimized to protect pictorial information.

It is well known that images are different from texts in many aspects, such as high redundancy and correlation. The main obstacle in designing effective image encryption algorithms is the difficulty of shuffling and diffusing such image data by traditional cryptographic means.

In most of the natural images, the value of any given pixel can be reasonably predicted from the values of its neighbours. Chaos based cryptosystem provides an efficient way to achieve image encryption. In the proposed block encryption/decryption algorithm, a 2D chaotic map is used to shuffle the image pixel positions.

For image encryption, two-dimensional (2D) chaotic maps are naturally employed as the image can be considered as a 2D array of pixels. Traditional symmetric ciphers such as Advanced Encryption Standard (AES) are designed with good confusion and diffusion properties. Two properties can also be found in chaotic systems of pseudo-randomness and periodicity which means a dynamical system that has the same behaviour averaged over time as averaged over space, high sensitivity to initial conditions and parameters.

Confusion property obscures the relationship between the plaintext and the cipher text and diffusion dissipates the redundancy in the plain text by spreading it out over the cipher text.

II. PROPOSED SYSTEM

The algorithm developed provides a method for purpose of encrypting and decrypting the image of any size and shape. It allows the user to select an image of his choice from a specified location on the computer, external hard drive or any other hardware devices connected to the computer. The system is able to support all standard image formats (e.g.:-TIFF, JPEG, BMP…..).The image selected by the user could be a Square image or a rectangular image of any dimension. The user is able to apply encryption to images captured via the camera and Personal pictures.

The image selected should be a colour image where the pixels are represented in the RGB model. Each pixel should be represented using minimum 24 pixels.

Once the keys have been entered by the user in any form, a standard chaotic map is generated. The chaotic map generated using Mathematical equations and theory is completely reversible, efficient enough to produce diffusion on the entire image pixels and the computation time is less. The chaotic map produced is then used for diffusing the image pixels. The image obtained from this chaos is completely distorted and the output is not recognizable by the end user.

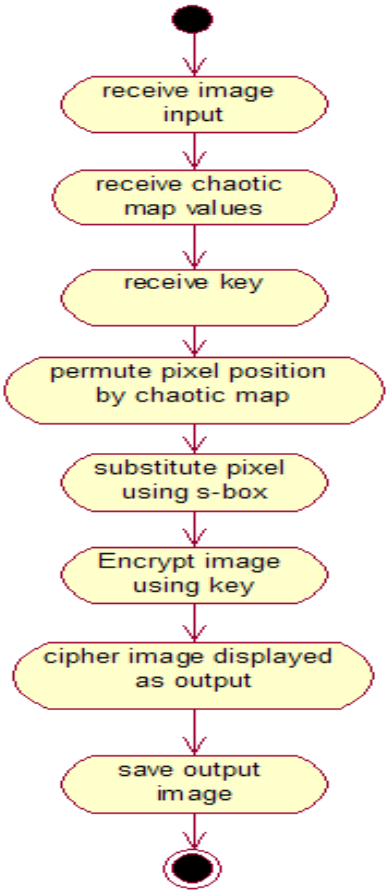

Fig 1. Sequence of steps

Approach 1: Sliding Window

In order to provide chaos to rectangular images a sliding window approach is used where a fixed square window should run on the image and all the pixels within the range of the window is shuffled. The window is then shifted by one column to produce diffusion on further part of the image.

Approach 2: Perfect Square

Another approach that has been used is that the rectangular image is converted into square image and then diffusion through chaotic map is applied on its pixels. The size of the original image is retained during the process of decryption.

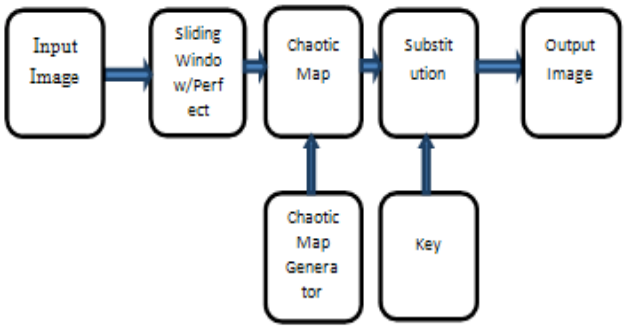

Fig 2. Block Diagram of CBCS (Chaos-Based Cryptosystem)

The Chaotic map used will be Arnold cat method which is a 2D chaotic map used for Cartesian coordinate system.

Equation of Arnold cat map is:

x=mod (2*i+j, m) +1;

y=mod (i+j, m) +1;

Where

x,y are new coordinated of pixels

i,j are original coordinated

m is size of square image

Then,the substitution phase will be basically where key is xored with the image pixel value. The xor operation occurs on pixels RGB values, so this will cause change in colour. Hence continuous colour of image is hidden.

III. GENERATION OF KEYS

The image can be encrypted using two types of keys.

A. Strict Key

In the strict key encryption algorithm the user is asked to enter a 32 digit hexadecimal number. This 32 digit hexadecimal number is then converted into a 128 bit binary key. Each hexadecimal key is represented by four bits. Since a 32-bit long key is entered by the user it corresponds to 128 bit binary key (32 X 4=128).This 128 bit key is then used for encrypting the pixels. Each 128 bit key has the potential to encrypt approximately 5 pixels. The image pixels are represented using RGB model. Each colour in the RGB model is represented using 8-bits.So a total of 24-bits are used to represent each pixel. So the 128 bit key can encrypt 128/24~5 pixels. The pixel encryption always starts from topmost first pixel and the encryption is performed from left to right.

The encryption is performed using Bit-XOR operation:-

The XOR gate with inputs *A* and *B* implements the logical expression $A.\overline{B} + \overline{A}.B$

After performing the encryption on neighbouring 5 pixels the 128 bit key is right-round rotated by one. The new key obtained is also 128 bit in size and is then used to encrypt the next 5 pixels. This process is continued and at each step the key is round rotated by one and then used to encrypt neighbouring pixels.

B. QUICK KEY

In this algorithm the user is asked to enter four decimal keys, say w, x, y, z. The four decimal keys entered should be less than 256.Since the maximum decimal number that can be entered is 255 its binary representation should consist of 8 bit binary digits. Each decimal key entered by the user is represented by 8 bits and the user enters four keys so we get a total of 32 bit key (8X4=32 bit). Now we perform rotate left operation on each (w, x, y and z separately) 8 bit key to obtain a four new 8 bit binary key.

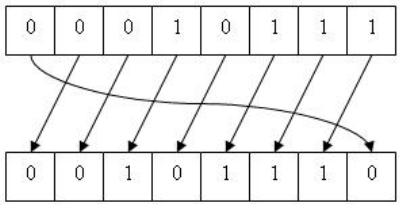

Fig 3. Key left rotate in CBCS(Chaos-Based Cryptosystem)

Now we perform rotate right operation on original four 8 bit binary key (w, x, y and z combined together)

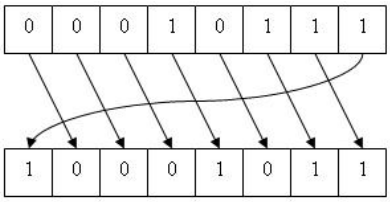

Fig 4. Key right rotate in CBCS(Chaos-Based Cryptosystem)

Next we perform XOR operation using this left shifted key and the right shifted key to obtain a new 32-bit binary key. Example: **0101 (decimal 5) XOR 0011 (decimal 3) = 0110 (decimal 6)**

We combine this four 32 bit key to obtain a 128 bit binary key. This 128 bit key is then used for encrypting the pixels. Each 128 bit key has the potential to encrypt approximately 5 pixels. The image pixels are represented using RGB model. Each color in the RGB model is represented using 8-bits.So a total of 24-bits are used to represent each pixel. So the 128 bit key can encrypt 128/24~5 pixels. The pixel encryption always starts from topmost first pixel and the encryption is performed from left to right.

After performing the encryption on neighbouring 5 pixels the 128 bit key is right-round rotated by one. The new key obtained is also 128 bit in size and is then used to encrypt the next 5 pixels. This process is continued and at each step the key is round rotated by one and then used to encrypt neighbouring pixels.

IV. DECRYPTION

The decryption process will be the reverse of encryption. First reverse substitution using the same key, so that colour is again readjusted and we get original colour. Next apply reverse chaotic map.

Equation for reverse Arnold cat map method is

x=mod (i-j-1, m) +1;

y=mod (2*j-i-2, m) +1;

Where

x,y are new coordinated of pixels

i,j are original coordinated

m is size of square image

Finally last stage in decryption process is resizing of image using reverse perfect square approach. In case of perfect square approach the image is converted to original rectangular shape .Whereas in case of Sliding window approach the image is

scanned from right to left. At each step we apply reverse chaotic map.

V. ASSUMPTIONS

The algorithm assumes that the image is represented using jpeg or gif format. All the pixels are 8 bit values.
The images should be represented using RGB colour model.

VI. CONSTRAINTS

The project requires the image to be converted to bitmap representation because the bit manipulation on pixel values has to be applied. The algorithm is calculation intensive, therefore high speed processors are required.

Speed and reliability of algorithm depends on the selected key.

Time complexity of the algorithm mainly depends on the speed of processor.

Space complexity of this algorithm is very high because of the huge image size and bit level manipulation of the pixels.

VII. RESULT

The algorithm will accept and image as input data.

For representations purpose we are using standard image for image processing.

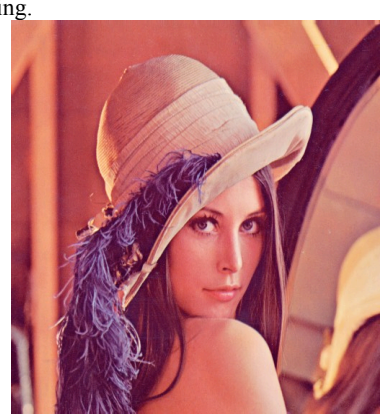

Fig 5. Image Input to CBCS

Then we apply the proposed algorithm,we get following output.

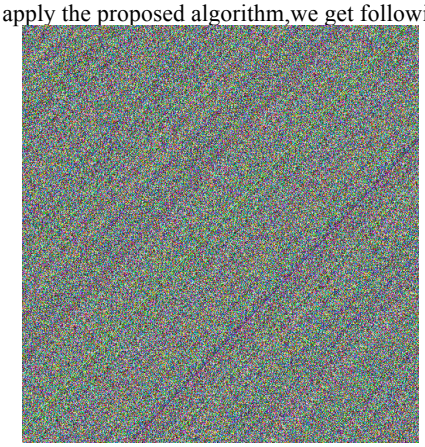

Fig 6. Image Output From CBCS

It should be noted that

1. Pixel positions on the image are shuffled due to Arnold cat chaotic map formula.
2. Colour of the pixel is also changed due to xor operation with key.

VIII. FUTURE SCOPE

The Project can be extended in such a way that it can have various future applications. The Project can be extended for encrypting videos. Since videos comprises was a sequence of frames. And each frame can be considered as an image.

Therefore this algorithm can be extended to encrypt a series of images which form a video. But the encryption should be performed on videos which have lossless format like MPEG-4 so that there is no loss of data. The project can also be made more robust and secured by using public key cryptography. In the proposed system, a symmetric key is used for encrypting the images.

Public-key cryptography refers to a cryptographic system requiring two separate keys, one to lock or encrypt the plaintext, and one to unlock or decrypt the cipher text. Neither key will do both functions. One of these keys is published or public and the other is kept private. If the lock/encryption key is the one published then the system enables private communication from the public to the unlocking key's owner.

If the unlock/decryption key is the one published then the system serves as a signature verifier of documents locked by the owner of the private key. This will help to increase the security of the system.

Moreover the project can be extended to create a secure channel to transfer the image from the client to the server by the software itself.

IX. CONCLUSION

The proposed system will work efficiently for image encryption and will provide several advantages over existing systems.

Advantages offered are:

1. User Flexibility:-The algorithm provides user flexibility by providing Encryption to a wide range of images. The software provides the flexibility of choosing standard format images of any shape and size.

2. Efficiency:-The computation time for encrypting is manageable using minimum requirements. Hardware having higher processing power takes less time to encrypt and decrypt the image.

3. Lossless Encryption and Decryption:-The algorithm provides Lossless encryption and decryption. There is no data loss during the process of encryption and decryption. Image quality is also maintained.

4. Two levels of security:-The algorithm provides two levels of security. First by producing diffusion using chaotic map. The second level of security is provided by using substitution of these image pixels.

X. REFERENCES

[1] G. Millérioux, J. M. Amigo, J. Daafouz, ―A connection between chaotic and conventional cryptography,‖ IEEE Trans. Circuits and Systems, vol. 55, no. 6, pp. 1695-1703, Jul. 2008.

[2] H. Xiao, S. Qiu, C. Deng, ―A Composite Image Encryption Scheme Using AES and Chaotic Series,‖ First International Symposium on Data, Privacy and E-Commerce, pp. 277279 –277279, 2007.

[3] A. Awad, A. Saadane, ―Efficient Chaotic permutations for image encryption algorithms‖, IAENG, International Conference of Signal and Image Engineering, pp. 748–753, 30 Jun-3 July , London, UK, 2010.

[4] S. Tao, W. Ruli, Y. Yixun, ―Perturbance based algorithm to expand cycle length of chaotic key stream,‖ IEEE, Electronics Letters, vol. 34, no. 9, pp. 873-874, 1998.

[5] T. Yang, C. W. Wu, L. O. Chua, ―Cryptography Based on Chaotic Systems,‖ IEEE Trans. Circuits and Systems, vol. 44, no.5, pp. 469–472, Feb. 1997.

[6] G. Jakimoski, L. Kocarev, ―Chaos and Cryptography: Block Encryption Ciphers Based on Chaotic Maps,‖ IEEE Trans. Circuits and Systems, vol. 48, no. 2, pp. 163–169, Feb. 2001.

[7] Security in Computing - Charles P. Pfleeger , Pearson Education.

[8] Cryptography and Network Security by Behrouz A. Forouzan, TATA McGraw hill.