International Journal of Advanced Information Science and Technology (IJAIST) ISSN: 2319:268
Vol.2, No.9, September 2013 DOI:10.15693/ijaist/2013.v2i9.6-11

# A Full Image Encryption Scheme Based on Transform Domains and Stream Ciphers

Sapna Anoop
Software Engineer
iGATE Information Services Pvt. Ltd.
Hyderabad, India

Anoop Alakkaran
Senior Consultant
Capgemini
Netherlands

***Abstract*—Encryption is the process of transforming information (plaintext) using an algorithm (called cipher) to make it unreadable to anyone except those possessing special knowledge, usually referred to as a key. A stream cipher is a fast symmetric key algorithm which is used to convert plaintext to ciphertext. Encryption of images can be done in both spatial and transform domain. A partial image encryption scheme based on DWT and Chaotic stream ciphers already exists. Here, a full image encryption scheme based on DWT and Stream Ciphers is proposed. A competitive study on image encryption schemes in transform domains (DCT, DWT) is also discussed in this paper.**

***Index terms –Selective Encryption, DCT, DWT, Stream Ciphers, PSNR***

## I. INTRODUCTION

The development of Information Technology and the rapid growth of computer networks allowed large files, such as digital images, to be easily transmitted in open networks such as the Internet. Each type of data has its own aspects, and different techniques should be used to protect confidential image data from unauthorized access. Security is an important issue in communication of images, and encryption is one of the ways to ensure security. Image encryption has applications in internet communication, multimedia systems, medical imaging, telemedicine, military communication etc. The encryption process requires an encryption algorithm and a key. The process of recovering plaintext from ciphertext is called decryption. The accepted view amongst cryptographers is that the encryption algorithm should be published, whereas the key must be kept secret (Kerckhoffs' law). A major recent trend is to minimize the computational requirements for secure multimedia distribution by selective encryption (SE) where only parts of the data are encrypted [1]. Selective encryption aims at avoiding the encryption of all bits of a digital image and yet ensuring a secure encryption [2]. Internet users exasperate potential security threats such as eavesdropping and illegal access. They want to be protected to ensure their privacy.

Encryption using stream cipher (RC4) in the spatial domain and in the transform domain is explained in Section II. Factors which are used for comparing the encryption schemes are measured in Section III.

## II. ENCRYPTING SPATIAL AND TRANSFORM DOMAIN DATA

The spatial domain is the normal image space, in which a change in position in images (I) directly projects to a change in position in scene (S) which might be a two-or three-dimensional. Transform coding relies on the premise that pixels in an image exhibit a certain level of correlation with their neighboring pixels. A transformation is, therefore, defined to map this spatial (correlated) data into transformed (uncorrelated) coefficients. Two types of transformations are Discrete Cosine Transform and Discrete Wavelet Transform.

A stream cipher is a symmetric key cipher where plaintext bits are combined with a pseudorandom cipher bit stream (keystream), typically by an exclusive-or (XOR) operation. In a stream cipher, the plaintext digits are encrypted one at a time. The main advantage of the stream cipher is that it is faster and more suitable for streaming application. RC4 is a stream cipher which is used for encryption in this paper. For encrypting the spatial domain data, a full encryption (FE) scheme is used which encrypts every pixel in the image using RC4 algorithm. The experimental results are shown in Section 3.1.

### *1) Encrypting Discrete Cosine Transform Data*

The Discrete Cosine Transform (DCT) is a mathematical transformation that takes a signal and transforms it from spatial domain into frequency domain [3]. Many digital image and video compression schemes use a block-based DCT, because this algorithm minimizes the amount of data needed to recreate a digitized image. In particular, JPEG (Joint Photographic Experts Group) and MPEG (Moving Picture Experts Group) use the DCT to concentrate image information by removing spatial data redundancies in two-dimensional images [4]. For encrypting the transform domain data, selective encryption is used. The concept behind encrypting only some selective DCT coefficients (the coefficients [0,0], [0,1], [0,2], [1,0], [2,0], [1,1]) is based on the fact that the image details are situated in the higher frequencies and the human is most sensitive to the lower frequencies than to the higher frequencies. An extra security has been provided to the encrypted blocks by shuffling the resulted blocks using the Shuffling Algorithm. The experimental results are shown in Section 3.2.

***Algorithm to Encrypt Image***

**Input :** Target Image to be encrypted and the stream RC4 Key values.
**Output :** Encrypted Image
**Begin**
**Step 1:** Read the image header, save the height of the image in variable height and the width in variable width and save the body image in an array image body.
**Step 2:** Obtain how many blocks exist in an image row and how many ones in the column, by dividing the width and

height of the image by N, where N is equal to 8 (the required block size).

NoRowB = Image Height / N;
NoColB = Image Width / N;

**Step 3:** For all blocks in the image perform the following:

Get_block (row_no, col_no)

Perform a DCT on the block and save the resulted coefficients in an array.

Round the selected coefficients, convert the selected coefficients to 11 bits.

Encrypt the selected coefficients by XORing the generated bit stream from the RC4 + Key with the coefficient bits, the sign bit of the selected coefficients will not be encrypted.

Perform an Inverse Discrete Cosine Transform (IDCT) and get the new block values, the resulted values could be positive or negative values due to the encryption step.

**Step 4:** Apply the proposed shuffling algorithm on the resulted blocks to obtain the encrypted image.

**End**

***Algorithm to Decrypt Image***

**Input :** Target Image to be decrypted and the Encryption Key

**Output :** Original Image

**Begin**

**Step 1:** Read the image header, save the height of the image in variable height and the width in variable width and save the body image in an array image body.

**Step 2:** Obtain how many blocks exist in an image row and how many ones in the column, by dividing the width and height of the image by N, where N is equal to 8.

NoRowB = Image Height / N;
NoColB = Image Width / N;

**Step 3:** Reshuffle the block, since the shuffling algorithm generates the same row and column numbers to return the shuffled blocks into their original locations.

**Step 4:** For all blocks in the image perform the following:

Get_block (row_no, col_no)

Perform a DCT on the block and save the resulted values in an array.

Round the selected coefficients, convert the selected coefficients to 11 bits.

Decrypt the resulted bits by using the generated bit stream from the RC4 + Key, by performing an XOR operation, the sign bit of the selected coefficients will remain.

Convert the resulted bits into integer values, and join the sign (from the step above) with each integer, if the coefficient is negative multiply it by −1.

Perform an Inverse Discrete Cosine Transform (IDCT) and get the new blocks.

**Step 5:** Reconstruct the image to get the original image.

**End**

***Shuffling Algorithm***

**Input:** Key, number of blocks in the row (NoRows), number of blocks in the column (NoCols) and the resulted encrypted image saved in an array.

**Output:** A new shuffled image

**Begin**

for i = 0 to (NoRows × NoCols)
NewVal [I] = (key × i) mod (NoRows × NoCols)
endfor

k= 0
for i = 0 to (NoRows × NoCols)
MoveBlock (ImageBlk (NewVal [i]), ImageBlk [k])
k++
endfor

**End**

***2) Encrypting Discrete Wavelet Transform data***

A Discrete Wavelet Transform (DWT) maps the time-domain signal of f(t) into a real-valued time frequency domain and the signals are described by the wavelet coefficients [5]. Wavelets are mathematical functions that cut up data into different frequency components. Wavelet algorithms process data at different scales or resolutions. The wavelet transform carries out a special form of analysis by shifting the original signal from the time domain into the time-frequency, or, in this context, time-scale domain. The idea behind the wavelet transform is the definition of a set of basic functions that allow an efficient, informative and useful representation of signals. There is a wide range of applications for Wavelet Transforms. They are applied in different fields ranging from signal processing to biometrics etc. A partial image encryption scheme based on DWT and chaotic stream ciphers already exists.

A full image encryption scheme is proposed here based on DWT and Stream Ciphers. The shuffling algorithm used here is same as that of the algorithm used in DCT domain. In the proposed method, the image first goes through the single-level DWT resulting in four coefficient matrices; the approximation (ca), horizontal (ch), vertical (cv), and diagonal (cd) matrices [6] [7]. The lowest frequency sub-band is expressed in the matrix ca. The ca matrix will be encrypted as it holds most of the image's information using the RC4 Stream Cipher. For encryption, the RC4 key stream will be combined with the ca coefficients using the XOR operation. While encrypting this matrix alone will provide complete perceptual encryption, it would be possible for an attacker to gain information about the image from the other matrices. Therefore, the horizontal (ch), vertical (cv), and diagonal (cd) matrices will be shuffled [8]. The Shuffling Algorithm used in the DCT method is used here. The encrypted ca matrix and the shuffled ch, cv and cd matrices then undergo the Inverse Discrete Wavelet Transform (IDWT) to produce the encrypted image.

This method aims at reducing encryption time by only encrypting part of the image, yet maintaining a high level of security by shuffling the rest of the image. The experimental results are shown in Section 3.3.

***Algorithm to Encrypt Image***

**Input :** Target Image to be encrypted and the stream RC4 Key values.

**Output :** Encrypted Image

**Begin**

**Step 1:** Read the image header, save the height of the image in variable height and the width in variable width and save the body image in an array image body.

**Step 2:** Obtain how many blocks exist in an image row and how many ones in the column, by dividing the width and height of the image by N, where N is equal to 8 (the required block size).

NoRowB = Image Height / N;
NoColB = Image Width / N;

**Step 3:** For all blocks in the image perform the following:

Get_block (row_no, col_no)

Perform a DWT on the block and save the resulted coefficients in an array.
Round the selected coefficients, convert the selected coefficients to 11 bits.
Encrypt the selected coefficients by XORing the generated bit stream from the RC4 + Key with the coefficient bits, the sign bit of the selected coefficients will not be encrypted.
Perform an Inverse Discrete Wavelet Transform (IDWT) and get the new block values, the resulted values could be positive or negative values due to the encryption step.
**Step 4:** Apply the proposed shuffling algorithm on the resulted blocks to obtain the encrypted image.
**End**

***Algorithm to Decrypt Image***

**Input :** Target Image to be decrypted and the Encryption Key
**Output :** Original Image
**Begin**
**Step 1:** Read the image header, save the height of the image in variable height and the width in variable width and save the body image in an array image body.
**Step 2:** Obtain how many blocks exist in an image row and how many ones in the column, by dividing the width and height of the image by N, where N is equal to 8.

NoRowB = Image Height / N;
NoColB = Image Width / N;

**Step 3:** For all blocks in the image perform the following:

Get_block (row_no, col_no)

Perform a DWT on the block and save the resulted values in an array.
Round the selected coefficients, convert the selected coefficients to 11 bits.
Decrypt the resulted bits by using the generated bit stream from the RC4 + Key, by performing an XOR operation, the sign bit of the selected coefficients will remain.
Convert the resulted bits into integer values, and join the sign (from the step above) with each integer, if the coefficient is negative multiply it by −1.
Perform an Inverse Discrete Wavelet Transform (IDWT) and get the new blocks.
**Step 4:** Reshuffle the block, since the shuffling algorithm generates the same row and column numbers to return the shuffled blocks into their original locations.
**Step 5:** Reconstruct the image to get the original image.
**End**

## III. PERFORMANCE EVALUATION AMONG IMAGE ENCRYPTION BASED ON SPATIAL DOMAIN WITH STREAM CIPHERS, DCT DOMAIN WITH STREAM CIPHERS AND PROPOSED DWT DOMAIN WITH STREAM CIPHERS

The performance of image encryption schemes for 512×512 images are measured using Entropy (Measure of randomness) and Encryption time.
Entropy is measured as a statistical measure of randomness.
Encryption time is the main parameter used in Image Encryption techniques. Encryption time should be reduced to improve the performance of the technique.

### *3.1) Results of Image Encryption scheme based on spatial domain and stream ciphers*

In Table 1, high entropy values are obtained for encrypted images when tested with different test images.

TABLE 1: MEASURES OF ENCRYPTED IMAGES IN SPATIAL DOMAIN

| Test Image | Entropy | Encryption time (sec) |
|---|---|---|
| Barbara | 7.9993 | 89.85 |
| House | 7.9993 | 90.19 |
| Lena | 7.9992 | 90.64 |
| Airplane | 7.9993 | 89.12 |
| Baboon | 7.9993 | 91.43 |

Figure 1. Original Image

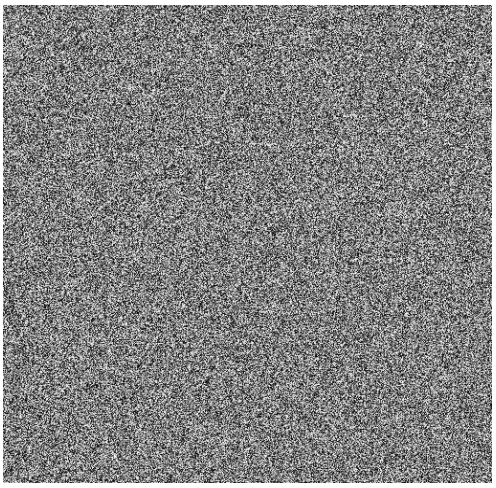

Figure 2. Encrypted Image of Figure 1

### *3.2) Results of Image Encryption scheme based on DCT and stream ciphers*

TABLE 2. MEASURES OF ENCRYPTED IMAGES IN DCT

| Test Image | Entropy | Encryption time (sec) |
|---|---|---|
| Barbara | 3.9693 | 35.58 |
| House | 3.9625 | 35.58 |
| Lena | 3.9654 | 36.59 |
| Airplane | 3.9571 | 36.64 |
| Baboon | 3.9470 | 34.94 |

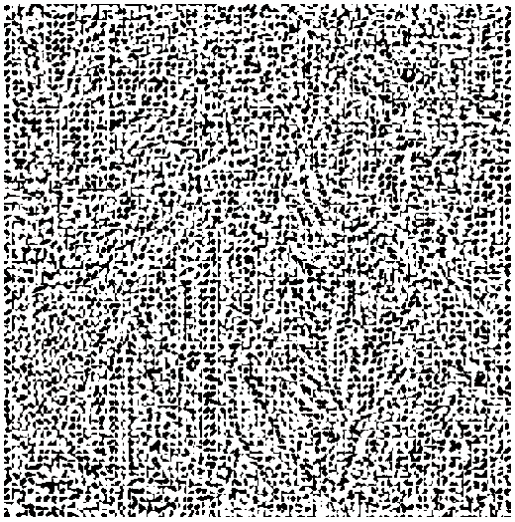

Figure 3. After Selective Encryption of Figure 1

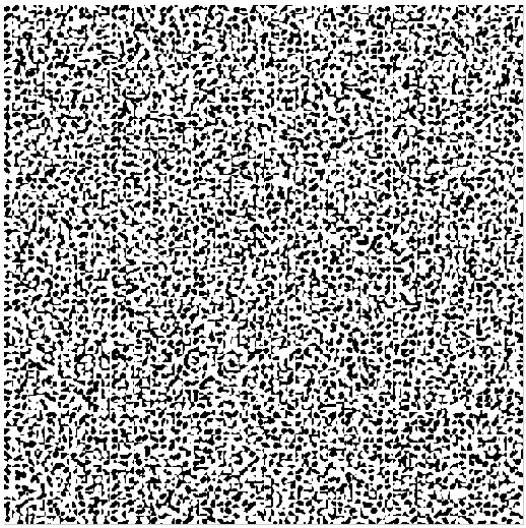

Figure 4. After Shuffling of Figure 3

### *3.3) Results of Image Encryption scheme based on DWT and stream ciphers*

TABLE 3. MEASURES OF ENCRYPTED IMAGES IN DWT

| Test Image | Entropy | Encryption time (sec) |
|---|---|---|
| Barbara | 5.7879 | 8.98 |
| House | 5.7888 | 8.92 |
| Lena | 5.7807 | 8.91 |
| Airplane | 5.7899 | 9.01 |
| Baboon | 5.7916 | 8.96 |

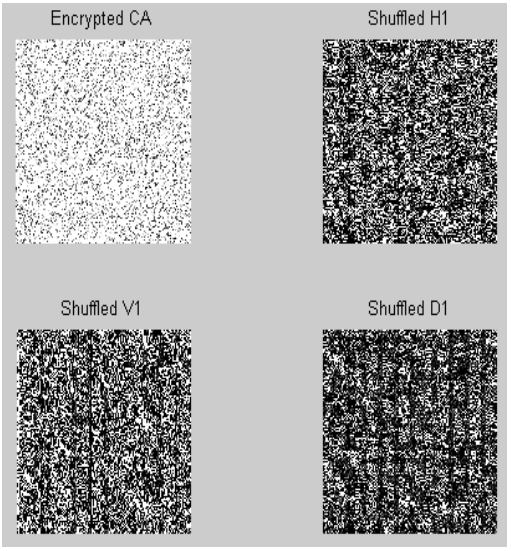


Figure 5. Encryption based on DWT (four coefficient matrices) of Figure 1

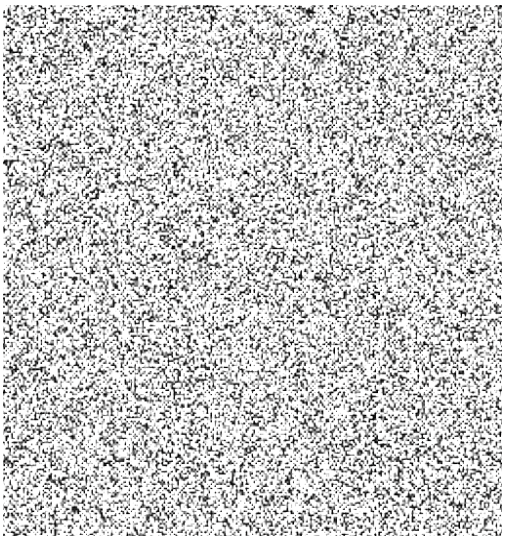

Figure 6. Full Encryption based on DWT of Figure1

Figure 7. Decrypted Image

***The below table shows the comparative PSNR values of Image Encryption scheme based on DCT with stream ciphers and DWT with stream ciphers***

PSNR is the Peak Signal-to-Noise Ratio.

TABLE 4. PSNR VALUES OF ENCRYPTED IMAGES FOR DIFFERENT TEST IMAGES

| Test Image | DCT | DWT |
|---|---|---|
| Barbara | 29.2989 | 20.5784 |
| House | 27.9692 | 20.7056 |
| Lena | 28.2182 | 20.8768 |
| Airplane | 29.5761 | 20.6219 |
| Baboon | 28.2258 | 20.7354 |

TABLE 5. PSNR VALUES OF DECRYPTED IMAGES FOR DIFFERENT TEST IMAGES

| Test Image | DCT | DWT |
|---|---|---|
| Barbara | 75.0756 | 75.6641 |
| House | 77.3615 | 75.4996 |
| Lena | 75.4900 | 75.5393 |
| Airplane | 75.0123 | 75.4515 |
| Baboon | 76.3480 | 75.3072 |

Comparison result shows that the PSNR values of encrypted images of DWT method are lower than the PSNR values of encrypted images in DCT domain method and the PSNR value of the decrypted image using DWT method is almost equal to DCT method. This shows the better performance of the DWT method as compared to DCT.

## IV. CONCLUSION

A full image encryption scheme based on DWT and Stream Ciphers have been presented in this paper and a competitive study on image encryption schemes in transform domains (DCT, DWT) is also discussed. DWT with stream cipher only encrypts the lowest frequency band of the image, however it is highly secure as the rest of the other bands are all shuffled using the Shuffling Algorithm. The algorithm is considered as a fast image encryption algorithm, due to the selective encryption of certain portion of the image (lowest frequency band). PSNR values of the encrypted images of DWT are low as compared to DCT and are resistant to statistical attacks. Comparatively higher entropy value is obtained in the case of DWT which implies a better performance. Encryption time should be lower and is obtained in the case of DWT than that of DCT. Hence, greater security is provided.

## REFERENCES

[1]. Lala Krikor, Sami Baba, Thawar Arif, Zyad Shaaban, "Image Encryption Using DCT and Stream Cipher", European Journal of Scientific Research, Vol.32, No.1, pp.47-57, 2009.

[2]. M. Van Droogenbroeck, R. Benedett, "Techniques for a Selective Encryption of Uncompressed and Compressed Images", in Proceedings of Advanced Concepts for Intelligent Vision Systems (ACIVS) 2002, Ghent, Belgium, September 2002.

[3]. G. Strang, "The Discrete Cosine Transform," SIAM Review, Volume 41, Number 1, pp.135-147, 1999.

[4]. C. Coconu, V. Stoica, F. Ionescu, D. Profeta, "Distributed Implementation of Discrete Cosine Transform Algorithm on a Network of Workstations", Proceedings of the International Workshop Trends & Recent Achievements in IT, Romania, pp. 116-121, May 2002.

[5]. Ramazan Gencay, Faruk Selcuk, Brandon Whitcher, "An Introduction to Wavelets and Other Filtering Methods in Finance and Economics", Academic Press, 2001.

[6]. Said E. El-Khamy, Mohammad Abou El-Nasr, Amina H. El-Zein, "A Partial Image Encryption Scheme Based on the DWT and ELKNZ Chaotic Stream Cipher", MASAUM Journal of Basic and Applied Sciences, Vol. 1, No. 3, October 2009.

[7]. S. Lian, Z. Wang, "Comparison of Several Wavelet Coefficients Confusion Methods Applied in Multimedia Encryption", In Proc. Int. Conference on Computer Networks and Mobile Computing (ICCNMC'2003), pp. 372–376, 2003.

[8]. G. Ginesu, T. Onali, D.D. Giusto, "Efficient Scrambling of Wavelet-based Compressed Images: A comparison between simple techniques for mobile applications", Proceedings of the 2nd International Mobile Multimedia Communications Conference (MobiMedia'06), 2006.

## Authors Profile

**Sapna Anoop** (previous name in International Journals and IEEE papers: **Sapna Sasidharan**) received her B.Tech degree in Computer Science from Sree Narayana Guru College of Engineering and Technology, Kerala. She has completed her M.Tech degree in Cyber Security from Amrita Vishwa Vidyapeetham University, Coimbatore. Her research interests are Image Encryption, Steganography and Cryptography. She is currently working as a Software Engineer in iGATE Information Services Pvt. Ltd., Hyderabad. She has

published 3 papers in International Journals and 3 IEEE Conference papers.

**Anoop Alakkaran** received his B.Tech degree in Electronics and Communication Engineering from TKM College of Engineering, Kerala University. His research interests are Image Encryption and Cryptography. He is currently working as a Senior Consultant in Capgemini, Netherlands.