International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Image Encryption-Compression Using Combined Permutation and SPIHT Algorithm

Jisha Shaji[1], Bijin Bodheswaran[2]

[1]PG Scholar, Sree Buddha College of Engineering for Women, Elavumthitta

[2]Assistant Professor, Sree Buddha College of Engineering for Women, Elavumthitta

Abstract: ***Encryption-Compression techniques are recently used for the effective transmission of images. By choosing the encryption and compression technique effectively, the efficiency of the system can be increased. Here combined permutation is applied for encryption and SPIHT algorithm is used for compression. The intelligible information is present in an image due to the correlation of bit, block and pixel. If we decrease this correlation then it will be difficult to understand the content. Wavelet based image compression using SPIHT algorithm is efficient and computationally simple. Matlab R2013a is used for performing this task. Both PSNR value and compression ratio are calculated for two images named satellite.jpg and lena.jpg. Results indicate that for satellite.jpg CR is 9.4660 and PSNR is 36.17dB and for lena.jpg it is 26.22 and 36.76 respectively. The results are present and discussed in the paper.***

Keywords: Encryption, Compression, Combined Permutation, SPIHT, PSNR.

1. Introduction

With the development of multimedia application and network technology, the security requirements become more and more important. Since, the multimedia network is transmitted over open network one by one, reliable security is required to protect the content of data from unwanted disclosure. The encryption technique needed for the multimedia data can be broadly classified into two. Encryption for real time video required lower level of security, since the classical cipher require heavy computation. But for government requirements and military application higher level of security is required. So depending upon the needs the level of encryption varied.
Next requirement is how effectively transmit the data through network. Traditionally, compression is done prior to encryption. The main demerit of this method is, if the transmitter does not use a resource deprived device, it will be difficult to done compression prior to encryption. Also, the channel provider has a tendency to compress the data in order to reduce the network traffic and the increase the network utility [1]. So the task for compression can be completely given to network provider. The main challenge in Encryption-Compression system is the network provider is unaware about the encrypted domain. A solution to this problem is the use of stream cipher; it is compressible through the use of coding with side information principle without compromising either the compression efficiency or the information theoretic security [2].

There are mainly two methods for compressing the data. Lossy compression and lossless compression. In the case of lossless compression, the reconstructed image is exactly resembles the original image without any loss of information. While in the case of lossy compression reconstructed image contains degradations with respect to original image. Another type of compression technique recently used is wavelet compression. Here, image is decomposes into a set of different resolution sub-images corresponding to the various frequency bands.

Here, the colour image is encrypted using combined permutation of block, bit and pixel and compression is done using SPIHT algorithm. The reverse process is carried out to extract the original image at the receiver. Figure 1 shows the block diagram of the entire system.

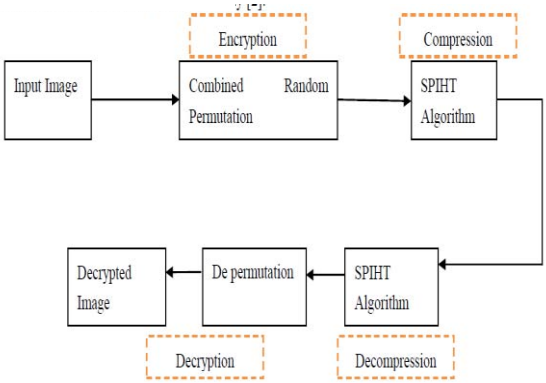


Figure 1: Block diagram of proposed system#

2. Image Encryption using Combined Permutation

An image can be viewed as an arrangement of bits, pixel and blocks. The intelligible information present in an image is due to the correlation among these parameters. This correlation can be reduced by permuting the block, bit and pixels. If we are permuting only the blocks, then it will be easy to understand [3]. So along with block, bits and pixels are permuted to obtain the encrypted image.

Block Permutation:

The image can be decomposes into various blocks. A group of block is selected and permuted using a key. The keys are generated from the pseudo random index generator. For better encryption the size of block should remain small. If it is too small, then it becomes difficult to encrypt the edges. At the receiver the reverse process is carried out.

Bit Permutation:
An image is a combination bits, bits are being taken from the image and permuted using the key generated from the pseudo random index generator. The reverse process is carried out in the receiver end to retrieve the image.

Pixel Permutation:
In this technique, a group of pixel is taken and permuted using the key as do with the block and bit. The size of key and pixels in the group should remain same. The reverse process is carried out in the receiver end.

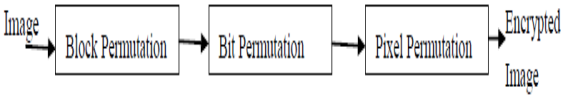


Figure 2: Combined Permutation Process

The permutation should be carried out in the order of block, bit and pixel and de permutation in pixel, bit and block. If this order is varied, the image will be in non-perceivable form.

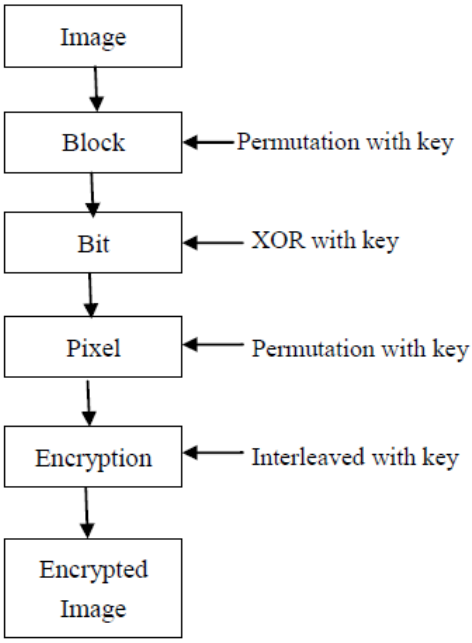


Figure 3: Flow chart of combined permutation

Algorithm for Combined Permutation
1. Read an RGB image.
2. Find the number of rows and columns; divide the image into small cells.
3. Apply permutation on this cell using the key generated from pseudo random index generator. Now convert back cells into matrix
4. Convert the block permuted image into bits.
5. The bits are XOR with the set of keys generated from pseudo random index generator.
6. Next is pixel permutation, a digital signature is generated using hash function. The random number generated from pseudo random index generator is used as the key for hash function.
7. Pixels are hashed. Here, MD5 hash function is used.
8. The pixel permuted image is then interleaved using the same key to obtain the encrypted image.

3. Set Partitioning in Hierarchical Tree (SPIHT)

Set Partitioning in Hierarchical Tree (SPIHT) algorithm employs spatial orientation tree and set partitioning sorting algorithm [4]. It is based on the fact that execution path of any algorithm is depend on the comparison of results of the branching points. So, if the encoder and decoder have the same sorting algorithm, then the decoder can duplicate the encoder's execution path if it receives the results of the magnitude comparisons, and the ordering information can be recovered from the execution path.

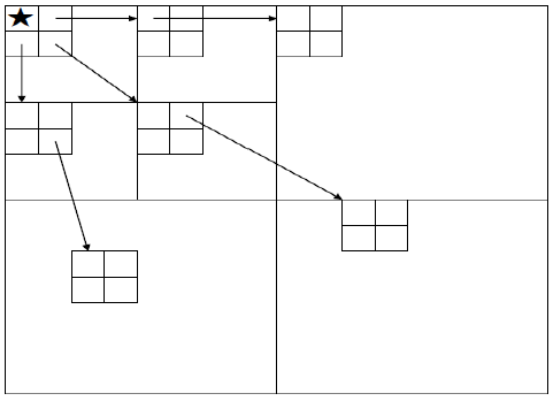


Figure 4: Parent offspring dependence in spatial orientation tree

The SPIHT algorithm mainly depends upon three lists LIP, LIS, LSP.

List of Insignificant Pixels: It contains individual pixels that having magnitude smaller than threshold.

List of Insignificant Set: It contains set of coefficients that are defined by the tree structure and having magnitude less than threshold.

List of Significant Pixels: The set of pixels that having value greater than the threshold. These pixels are not further evaluated.

SPIHT consists of four pass: Initialization pass, Sorting pass, Refinement pass, Quantization-step update.

The following set of coordinated are used in the algorithm:
O(i,j): The set of coordinates of all offspring of node *(i,j)*
D(i,j): The set of coordinates of all descendants of node *(i,j)*
H: The node in the highest level.
L(i,j)=D(i,j)-O(i,j).

3.1SPIHT Algorithm

1. INITIALIZATION PASS

$$\text{output} n = \left\lfloor \log_2 \left(\max_{(i,j)} \left\{ \left| c_{i,j} \right| \right\} \right) \right\rfloor;$$

Set LSP = empty set; LIP = coordinates of (i,j); type A = those with descendants also in LIS.
2. SORTING PASS
(i) For each entry (i, j)in LIP do:
a) Output $S_n(i, j)$;

(ii) For entry (i, j) in LIS do:
a) If entry is of type A then
Output $S_n(D(i, j))$;
If $S_nD(i, j) =1$ then
For each (k,l)Ɛ O(i, j) do:
Output $S_n(k, l)$;
b) If $S_n(i, j)=1$, move to LSP and output sign of $C_{i,j}$
If $S_n(k, l) =1$, then add (k,l) to the LSP and output sign of $C_{k,l}$
If $S_n(k, l) =0$, then add (k, l) to the end of LIP.
If $L(i, j)\neq0$, then move (i, j) to the end of the LIS, as an entry of type B and go to the next step (b) otherwise remove entry (i, j) from LIS.
b) If the entry is of type B then
Output Sn(L(i, j));
If Sn(L(i, j)) =1 then, Add each (k,l)Ɛ O(i, j) to the LIS as a type A and remove (i, j) from LIS.
3. REFINEMENT PASS: For each entry (i, j) in LSP except those included in the last sorting pass, output the n^{th} most significant bit of $|C_{i,j}|$;
4. QUANTIZATION STEP UPDATE:
Decrement n by 1 and go to step 2 [5].

4. Results

Matlab R2013a is used as the platform to perform this task. The RGB image is first read and displayed. Then combined permutation is carried out using key. This key vector will act as channel side information, so that the compression can be done in the encrypted domain.

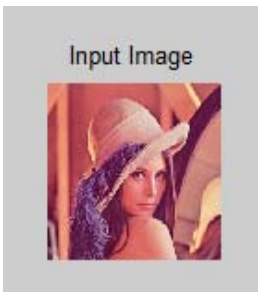
Figure 5 shows the RGB image chosen. Here lena.jpg is selected. Second figure shows the result after performing combined permutation. Then the image is transmitted. At the channel compression is carried out using SPIHT and then transmitted to the receiver.

The encryption technique used in this paper is more secure. If the order and process is changed then it will be difficult to find exact output.

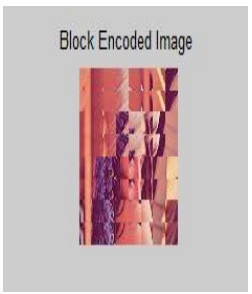
Table 1: Results obtained

Figure Name	Compression Ratio	PSNR
satellite.jpg	9.466	36.17
Lena.jpg	26.22	36.76

From the table above shows compression ratio and PSNR value of satellite.jpg and lena.jpg. It is clear that, the compression ratio is in the range of 30-37 for SPIHT algorithm.

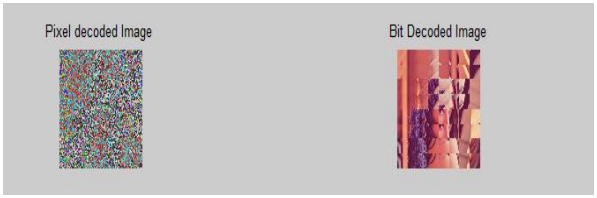


a)

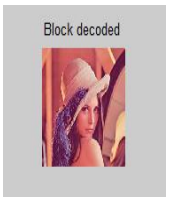


b)

c) d)



e) f)



g)

Figure 5: Results. a) Original Image b) Block Encoded Image c) Bit Encoded Image d) Pixel Encoded Image e) Pixel Decoded Image f) Bit Decoded Image g) Block Decoded Image.

5. Conclusion

In this paper, encryption then compression technique is used. The encryption is carried out using combined permutation, and compression using SPIHT algorithm. The reverse process is carried out in the receiver side to get the image back. Two set of images are used for the analysis of PSNR and compression ratio. First is satellite.jpg, it produces a result of 36.17dB and 9.466 as PSNR and CR respectively. While in the case of lena.jpg this values are 36.76dB and 26.22 respectively. Hence we can conclude that it is an efficient method.

References

[1] J. Zhou and X. Liu, "Designing an efficient image encryption-then-compression via prediction error clustering and random permutation", IEEE Transaction on Information Forensics and Security, vol.9, Jan. 2014.
[2] M. Johnson and P. Ishwar, "On Compressing an encrypted image", IEEE Trans. Signal Process., vol. 52, pp. 2992-3006, Oct. 2004.
[3] A. Mitra *et. al.,* "A new image encryption approach using combinational permutation technique", International journal of electrical and computer engineering, Feb.2006
[4] Ritu Chourasiya and Prof. Ajith Shrivastas, "A study of image compression based transmission algorithm using SPIHT for low bit rate application", Advanced computing: An International Journal, vol.3,Nov.2012.
[5] A. Said and W.A. Pearlman, "A new, fast and efficient image codec based on Set Partitioning in Hierarchical

Tree", IEEE Trans. On Circuits and Systems for Video Technology, vol.6, June 1996.
[6] www.mathworks.in

Author Profile

Jisha Shaji is pursuing her M.Tech degree in Communication Engineering from Sree Buddha college of Engg. For women, Elavumthitta, Pathanamthitta.

Bijin Bodheswaran working as Assistant Professor in department of Electronics and Communication, Sree Buddha college of Engg. For women, Elavumthitta, Pathanamthitta.