Efficient Techniques For Partial Encryption of Wavelet-based Compressed Digital Images

Dr. Hameed A. Younis, Dr. Turki Y. Abdalla**, Dr. Abdulkareem Y. Abdalla**

**Dept. of Computer Science, College of Science, University of Basrah, Basrah, Iraq.*
***Dept. of Computer Engineering, College of Engineering, University of Basrah, Basrah, Iraq.*

Abstract

The use of image communication has increased in recent years. In this paper, new partial encryption schemes are used to encrypt only part of the compressed data. Only 0.0244-20.4357% of the original data is encrypted for four different images, resulting in a significant reduction in encryption and decryption time. In the compression step, the advanced wavelet coding schemes (the Embedded Zerotree Wavelet (EZW) algorithm, the Set Partition in Hierarchical Trees (SPIHT) algorithm) and Joint Photographic Expert Group (JPEG) technique are used. In the encryption step, the Advanced Encryption Standard (AES) cipher is used. The effect of different compression ratios is studied. The proposed partial encryption schemes are fast and secure, and do not reduce the compression performance of the underlying selected compression methods.

Keywords: Image encryption, Compression, AES cipher, SPIHT, EZW, JPEG.

تقنيات كفوءة للتشفير الجزئي معتمدة على التحويل الموجي للصور الرقمية المضغوطة

د. حميد عبد الكريم يونس*، د. تركي يونس عبد الله**، د. عبد الكريم يونس عبد الله*

*قسم علوم الحاسبات، كلية العلوم، جامعة البصرة، البصرة، العراق.

**قسم هندسة الحاسبات، كلية الهندسة، جامعة البصرة، البصرة، العراق.

المستخلص:

ازداد الاهتمام في اتصالات الصور في السنوات الأخيرة. في هذا البحث، اقترحت طرقا للتشفير الجزئي الجديدة، والتي فيها تقوم خوارزميـة التشفير بتشفير جزء من البيانات المضغوطة. وشفر بحدود (0.0244-20-4357%) من البيانات الأصلية المستخدمة (أربع صـور مختلفـة) للحصول على تقليل مهم في زمن التشفير وفك الشفرة. استخدمت في مرحلة الضغط، تقنيات تحليل موجي متقدمة مثل خوارزمية الشجرة الصفرية المرتبة (EZW)، تقسيم المجموعة في اشجار هرمية (SPIHT) وتقنية JPEG. وفي مرحلة التشفير، استخدمت طرق تشفير متقدمة مثل التشفير القياسي المتقدم (AES). درست تأثير نسب ضغط مختلفة. أنظمة التشفير الجزئي المقترحة تكون سريعة وذات سرية عاليـة كمـا إن انجازية الضغط لا تقل ضمن طرق الضغط المختارة.

الكلمات المفتاحية: تشفير صورة، الضغط، JPEG, EZW, SPIHT, AES

1. Introduction

The use of image communication has increased dramatically in recent years. The World Wide Web and video conferencing are two examples. When communication bandwidth is limited, data is often compressed before transmission. If there is a need to protect the transmission from eavesdroppers, the transmission is also encrypted. For example, a wireless network often has limited bandwidth and its network traffic can easily be intercepted [1]. As a result, transmissions over a wireless network need to be compressed and encrypted. Traditionally, an appropriate compression algorithm is applied to the multimedia data and its output is encrypted by an independent encryption algorithm. This process must be reversed by the receiver.

Unfortunately, the processing time for encryption and decryption is a major factor in real-time image communication. In addition, the processing time required for compression and decompression of an associated image data is important. Encryption and decryption algorithms are too slow to handle the tremendous amount of data transmitted. Ciphering of images is actually an important issue. One essential difference between text data and image data is that the size of image data is much larger than the text data. The time is a very important factor for the image encryption. We find it at two levels, one is the time to encrypt, the other is the time to transfer images. To minimize the time, the first step is to choose a robust, rapid and easy method to implement cryptosystem. The other important criterion concerns the method of compression is that to decrease the size of images without loss of image quality [2].

Wavelet Transform (WT) is one of the most powerful tools in digital signal processing. The image components are decomposed into different decomposition levels using a wavelet transform. These decomposition levels contain a number of subbands, which consist of coefficients that describe the horizontal and vertical spatial frequency characteristics of the original image component [3]. Power of 2 decompositions are allowed in the form of standard decomposition.

To perform the forward Discrete Wavelet Transform (DWT), the standard uses a two dimension (2-D) subband decomposition of a 2-D set of samples into low-pass samples and high-pass samples. Low-pass samples represent a downsampled low-resolution version of the original set. High-pass samples represent a downsampled residual version of the original set, needed for the perfect

reconstruction of the original set from the low-pass set. It is mainly used to de-correlate the image data, so the resulting wavelet coefficients can be efficiently coded. It also has good energy compaction capability that results in a high compression ratio [4].

The aim of algorithm proposed here is to combine image compression with encryption. Many researchers have examined the possibility of combining compression and encryption [1, 2, 5, 6]. In this paper, we propose several approaches of partial encryption to reduce encryption and decryption time in image communication [7]. In these approaches, only part of the compressed data is encrypted

2. Basic Principles

2.1 EZW Algorithm

An EZW encoder was especially designed by Shapiro [8] to be use with wavelet transforms. In fact, EZW coding is a quantization method. The EZW encoder is based on progressive encoding to compress an image into a bit stream with increasing accuracy. This means that when more bits are added to the stream, the decoded image will contain more detail.

A zerotree is a tree of which all nodes are equal to or smaller than the root. The tree is coded with a single symbol and reconstructed by the decoder as a tree filled with zeroes [8, 9, 10].

In this method the Morton scan order was used. The first step in the EZW coding algorithm is to determine the initial threshold. The initial threshold t_0 will be:

$$t_0 = 2^{\lfloor \log_2(MAX(|\gamma(x,y)|)) \rfloor} \quad \ldots (1)$$

Here MAX(.) means the maximum coefficient value in the image and $\gamma(x,y)$ denotes the coefficient. With this threshold, the main coding loop is inserted:

```
threshold = initial_threshold;
do {
  dominant_pass(image);
  subordinate_pass(image);
  threshold = threshold/2;
    }while(threshold>
minimum_threshold);
```

It is noticed that two passes are used to code the image. In the first pass, the *dominant pass*, the image is scanned and a symbol is output for every coefficient. If the coefficient is larger than the threshold, a **P** (positive) is coded. If the coefficient is smaller than minus, the threshold an **Nv** (negative) is coded. If the coefficient is the root of a zerotree, then a **T** (zerotree) is coded, and finally, if the coefficient is smaller than the threshold but it is not the root of a zerotree, then a **Z**

(isolated zero) is coded. This happens when there is a coefficient larger than the threshold in the subtree. The effect of using the **Nv** and **P** codes is that when a coefficient is found to be larger than the threshold (in absolute value or magnitude), its two most significant bits are output (if we forget about sign extension).

The second pass, the *subordinate pass*, is the refinement one. After the dominant pass, the subordinate pass follows:

```
/*Subordinate pass*/
subordinate_threshold=
current_threshold/2;
for all elements on subordinate list do
{
 if (coefficient > subordinate_threshold)
  {
   output a one;
   coefficient=coefficient-subordinate_
threshold;
  }
 else output a zero;
}
```

2.2 SPIHT Algorithm

The quantization method used to generate some of the results in this paper is the Set Partitioning In Hierarchical Trees ***(SPIHT)*** which is developed by Said and Pearlman [11]. Said and Pearlman have significantly improved the Sapiro's EZW algorithm [8]. The SPIHT quantizer is an embedded coder that achieves good performance by exploiting the spatial dependencies in the subbands of the wavelet decomposition [11, 12]. The SPIHT coder was chosen for the experiments in this paper due to its good objective and computational performance.

For best understanding of how SPIHT works, the pixels relationship should be explained. In particular, each pixel in a smaller subband has four children in the next larger subband in the form of a 2×2 block of adjacent pixels. Each small square represents pixel and each arrow points from a particular parent pixel to its 2×2 group of children. The importance of the parent-child relation in quantization is described by the following statement: if the parent coefficient has a small value, then the children will most likely have small values. Conversely, if the parent has a large value, one or more of the children may also have large value.

Coders like SPIHT exploit this spatial dependence by partitioning the pixel values into parent-descendent groups. The coder starts with a threshold value that is the largest integer power of two. This power does not exceed the largest pixel value. Pixels are evaluated in turn to see if they are larger than the threshold; if not, these pixels are considered insignificant. If a parent and all of its descendents are insignificant, then the coder merely records the parent's coordinates. Since the

children's coordinates can be inferred from those of the parent, those coordinates are not recorded, resulting in a potentially great savings in the output bit stream. After locating and recording all the significant pixels for the given threshold, the threshold is reduced by a factor of two and the process repeated. By the end of each stage, all coefficients that have been found to be significant will have their most significant bits (when considered as binary integers) recorded [12, 13].

2.3 JPEG Technique

The JPEG/DCT still image compression has recently become a standard [14]. To exploit this method, an image is first partitioned into nonoverlapped 8×8 blocks. A DCT is applied to each block to convert the gray levels of pixels in the spatial domain into coefficients in the frequency domain. The coefficients are normalized by different scales according to the quantization table provided by the JPEG standard conducted by some psychovisual evidence. The quantized coefficients are rearranged in a zigzag scan order to be further compressed by an efficient lossless coding strategy such as run length coding, arithmetic coding or Huffman coding. The decoding is simply the inverse process of encoding. So, the JPEG compression takes about the same time for both encoding and decoding.

The information loss occurs only in the process of coefficient quantization. The JPEG standard defines a standard 8×8 quantization table for all images which may not be appropriate. To achieve a better decoding quality of various images with the same compression by using the DCT approach, an adaptive quantization table may be used instead of using the standard quantization table.

2.4 Permutation Cipher

In this system, the position of the plaintext letters in the massage rather than the letters of alphabet are permuted, while the permutation is the key. For the digital image the position of pixels are rearranged for different algorithms according to a key, such as image reversal, row transposition, column transposition, and block or matrix transposition [15, 16].

2.5 Advanced Encryption Standard (AES) Cipher

The AES cipher described by Rijndael (called also *Rijndael encryption algorithm*) [16, 17], it is a block cipher that converts cleartext data blocks of 128, 192, or 256 bits into ciphertext blocks of the same length. The AES cipher uses a key of selectable length (128, 192, or 256 bits). This encryption algorithm is organized as a set of iterations called *round transformations*. In each round, a data block is transformed by series of operations. The total number of rounds

depends on the largest of round r and key length kl, and equals 10, 12, and 14 for lengths of 128, 192, and 256 bits, respectively. All round transformations are identical, apart from the final one. The AES algorithm takes the cipher key, and performs a key expansion routine to generate a key schedule. For number of round = 10 and key length = 128 bits, the key expansion generates a total of 44 words. The resulting key schedule consists of a linear array of 4-byte words, denoted by [wi], with i in the range $0 \leq i < 44$.

3. The Proposed Techniques

3.1 SPIHT-AES Partial Encryption Scheme (SPIHT-AES-PE)

In this scheme, we propose a method for partial encryption of compressed image. The proposed method consists of wavelet transform (8 levels), quantization by SPIHT, encryption of important part then coding of resultant image by using run length coding.

The encryption step in this algorithm can be preformed by using any standard encryption algorithm. In the proposed scheme, a AES cipher is tested.

During the compression step, the SPIHT image coding algorithm is used, which can achieve a reasonably good compression rate. Among all wavelet-based image compression schemes, SPIHT quantization shows its remarkable performance not only in terms of efficiency but also in its low computational cost and progressive coding characteristics. Progressive coding (also called embedding coding) refers to the way that the most significant bits representing an image are placed at the beginning of the code, and the code bits are arranged according to their importance relative to the representation of the image. SPIHT quantizer is an embedded coder that the pixels are sorted descendently in the output bit stream according to the information importance. The important part is the first part of bitstream. The details of the SPIHT quantization is described in section (2.2).

In this scheme, only the important part of bitstream of image of SPIHT quantization is encrypted whereas the remaining parts (unimportant parts) are transmitted without encryption. The important part of the bitstream (small size 1%) is encrypted with the AES cipher (SPIHT-AES-PE).

SPIHT-AES-PE Algorithm:

1. Encryption key selection.
2. Wavelet filter selection.
3. Decomposition (filtering) the image, here discrete wavelet transform (8 levels) is used.

4. Quantization, here SPIHT quantization process is applied.

5. Partial encryption, here AES cipher is used.

6. Entropy coding, here the run length coding [7] is adopted.

3.2 EZW-Permutation Partial Encryption Scheme(EZW-Permutation-PE)

In this scheme, we propose to use the EZW quantization together with Permutation cipher. The method consists of wavelet transform (8 levels), EZW quantization, Permutation cipher and arithmetic coding [7].

EZW quantization process is based on progressive coding to compress an image into a bitstream with increasing accuracy. Two lists are used to code the image into a dominant list and subordinate one. The important part is the elements of the subordinate list. Only the first part of bitstream image of EZW quantization (subordinate list) (important part of size 20%) is encrypted with Permutation cipher, whereas the remaining part (unimportant part) is transmitted without encryption. The EZW quantization is described in section (2.1)

EWZ-Permutation-PE Algorithm:

1. Encryption key selection.

2. Wavelet filter selection.

3. Decomposition (filtering) the image, here discrete wavelet transform (8 levels) is used.

4. Quantization, here EZW quantization process is applied.

5. Partial encryption, here Permutation cipher is used.

6. Entropy coding, here the arithmetic coding is adopted.

3.3 JPEG-Permutation-AES Partial Encryption Scheme (JPEG-Permutation-AES-PE)

In this scheme, we propose a method for partial encryption of compressed image. It consists of discrete cosine transform, quantization by scalar quantization, encryption of important part, then coding of resultant image by using run length.

Because of high relationship between DC coefficients of neighbour blocks, we calculate the difference between each block of DC and the pervious one. The following equation shows this relationship after quantization:

$$DIFF_i = DC_i - DC_{i-1} \qquad \ldots(2)$$

Then, zigzag scan is considered.

We propose here to encrypt only the DC coefficients using Permutation cipher,

and then AES cipher. The part of blocks of DCT (DC coefficients) (important part) is put in the list. This list is encrypted through Permutation cipher. Then, AES cipher is adopted. The encrypted DC coefficients return to their positions in the blocks.

4. Experimental Results

In this section, a number of experiments which are used to examine our proposed algorithms will be presented. The algorithms were programmed in MATLAB version 6.5 on a Pentium IV PC (2.4 GHz) using four grayscale images of (256×256) pixels.

To evaluate each of the proposed schemes, five aspects are examined [1]:

1. **Security**. Security in this work means confidentiality and robustness against attacks to break the images. It is obvious that the goal is not 100% security, but many advanced algorithms are adopted, such as AES, and Stream ciphers that make them difficult to cryptanalyze.
2. **Speed**. Less data (important part) to encrypt means less CPU time required for encryption. So, in general partial encryption algorithms are used to reduce encryption and decryption time.
3. **Correlation**. Correlation (*Corr*) measures the similarity between the original image and the reconstructed image. The aim is to get a correlation value closed to 1.

The correlation can be defined as [1]:

$$Corr = \frac{\sum_{r=1}^{N}\sum_{c=1}^{M}(I_1(r,c)-\bar{I}_1)(I_2(r,c)-\bar{I}_2)}{\sqrt{[\sum_{r=1}^{N}\sum_{c=1}^{M}(I_1(r,c)-\bar{I}_1)^2][\sum_{r=1}^{N}\sum_{c=1}^{M}(I_2(r,c)-\bar{I}_2)^2]}} \quad \ldots(3)$$

Where:

$I_1(r,c)$: is the value of pixel at *(r,c)* of the original image.

$\bar{I}_1$: is the mean of the original image that:

$$\bar{I}_1 = \frac{1}{M \times N}\sum_{r=1}^{N}\sum_{c=1}^{M} I_1(r,c) \quad \ldots(4)$$

$I_2(r,c)$: is the value of pixel at *(r,c)* of the reconstructed image (or modified image).

$\bar{I}_2$: is the mean of the reconstructed image (or modified image) that:

$$\bar{I}_2 = \frac{1}{M \times N}\sum_{r=1}^{N}\sum_{c=1}^{M} I_2(r,c) \quad \ldots(5)$$

M: height of the image.

N: width of the image.

r and c: row and column numbers.

4. **PSNR**

Peak signal-to-noise ratio (PSNR) is the standard method for quantitatively comparing a compressed image with the original. For an 8-bit grayscale image, the peak signal value is 255. Hence, the PSNR of an M×N 8-bit

grayscale image x and its reconstruction $\hat{x}$ is calculated as [18]:

$$PSNR = 10\log_{10}\frac{255^2}{MSE} \quad \ldots (3)$$

where the Mean Square Error (MSE) is defined as:

$$MSE = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}[x(m,n) - \hat{x}(m,n)]^2 \quad \ldots (6)$$

PSNR is measured in decibels (dB), M: height of the image and N: width of the image.

5. **Compression Ratio (CR)**

The method of comparing the compressed and the original images is the compression ratio. It is defined as [19]:

$$Compression\ \ Ratio = \frac{Compressed\ \ File}{Uncompressed\ \ File} \quad \ldots (7)$$

The implementation consists of several experiments. Some of these experiments run with different CRs. In the following experiments, experimental results will be presented.

Experiment 1

In this experiment, SPIHT partial encryption scheme is considered. Four different CRs are chosen for this experiment, which are 1, 0.5, 0.25 or 0.125.

The important part of bitstream of SPIHT quantization image is encrypted by using AES encryption algorithm as follows:

SPIHT-AES-PE:

We propose here to encrypt important part by using AES cipher. Results obtained by applying this method are presented in Table (1). Figure (1) shows the results obtained for birds image.

In Table (1), the first column gives the CR. The second column gives the PSNR for each test grayscale image (Lena, house, birds or boys). The encryption key is "2b 7e 15 16 28 ae d2 a6 ab f7 15 88 09 cf 4f 3c". Only the first 128 bits (0.0244%) of the original data is encrypted for the test images.

Experiment 2

In this experiment, the proposed EZW-Permutation partial encryption algorithm is considered. Results obtained by applying this encryption method are presented in Table (2). Figure (2) shows results obtained for birds image.

In Table (1), the first column gives the grayscale test images. The second column gives the CR. Finally, the third column gives the PSNR of the reconstructed image. The encryption key contains positions of 38462 pixels which are randomly generated. Only the first 20.4357% of the original data is encrypted for the test images.

Experiment 3

In this experiment, the proposed JPEG-Permutation-AES partial encryption algorithm is considered. Results obtained

by applying this encryption method are presented in Table (3). Figure (3) shows the results obtained for birds image.

The encryption key involves positions of 1024 pixels randomly generated for Permutation cipher and "2b 7e 15 16 28 ae d2 a6 ab f7 15 88 09 cf 4f 3c" for AES cipher. Only 1.5625% of the original data is encrypted for the test images.

5. Conclusion

In this paper, three different partial encryption of images are proposed and tested. The first method, consists of 8 levels wavelet transform and SPIHT quantization with AES cipher. In the second method EZW quantization and permutation cipher are used. In the third method, we suggest to use discrete cosine transform, scalar quantization and permutation and AES cipher.

In all experiments, the attacker cannot obtain the original image unless he knows the encryption key. So, the proposed methods have good security since the keyspace is very large (2^{128}), so they have good security. The speed of our proposed algorithms is good since small size of the image is encrypted.

Out of the results of experiment 1, one can notice that as the CR increases, both execution time and PSNR value of the reconstructed image will increase. The average one can take is the second case (CR = 0.5). It is an acceptable one since it gives an acceptable PSNR and a reasonable time. Figure (4) shows PSNR versus CR for Lena image.

In EZW-Permutation method, high PSNR values are obtained, but the execution time is very long compared to other techniques. In this approach the image is a good quality.

6. References

[1] **Cheng H.,** ***"Partial Encryption for Image and Videc Communication",*** M.Sc. Thesis, Department of Computing Science, University of Alberta, Alberta, 1998.

[2] **Borie J., Puech W., and Dumas M.,** ***"Crypto-Compression System for Secure Transfer of Medical Images",*** 2nd International Conference on Advances in Medical Signal and Information Processing (MEDSIP 2004), September 2004.

[3] **Uehara T., Safavi-Naini R., and Ogunbona P.,** ***"Securing Wavelet Compression with Random Permutations",*** In Proceedings of the 2000 IEEE Pacific Rim Conference on Multimedia, pp. 332-335, Sydney, 2000.

[4] **Usevitch B. E.,** ***"A Tutorial on Modern Lossy Wavelet Image Compression: Foundations of JPEG 2000",*** IEEE Transactions on Image Processing Magazine, September 2001.

[5] **Li X., Knipe J., and Cheng H.,** ***"Image Compression and Encryption Using Tree Structures"***, Pattern Recognition Letters, Vol. 18, No. 11-13, pp. 1253-1259, 1997.

[6] **Norcen R., Podesser M., Pommer A., Schmidt H., and Uhl A.,** ***"Confidential Storage and Transmission of Medical Image Data"***, Computers in Biology and Medicine 33, pp. 277-292, 2003.

[7] **Younis, H. A.,** ***"New Techniques For Partial Encryption of Wavelet-based Compressed and Uncompressed Images"***, Ph.D. Thesis, Department of Computer Science, College of Science, University of Basrah, Basrah, November 2006.

[8] **Shapiro J. M.,** ***"Embedded Image Coding Using Zerotrees of Wavelet Coefficients"*** IEEE Transactions on Image Processing, Vol. 41, No. 12, pp. 3445-3462, December 1993.

[9] **Rajpoot N. , and Wilson R.,** ***"Progressive Image Coding Using Augmented Zerotrees of Wavelet Coefficients"***, Department of Computer Science, University of Warwick, Coventry, September 1998.

[10] **Valens C.,** ***"EZW Encoding"***, http://perso.wanadoo.fr/polyvalens/clemens/ezw.
E-mail: wavelets@polyvalens.com.

[11] **Said A., and Pearlman W. A.,** ***"A New Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees"***, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 6, No. 3, pp. 243-249, June 1996.

[12] **Morales A., and Agili S.,** ***"Implementing the SPIHT Algorithm in MATLAB"***, In Proceedings of the 2003 ASEE/WFEO International Colloquium, 2003.

[13] **Saha S.,** ***"Image Compression-From DCT to Wavelet: A Review"***, ACM Crossroads Student Magazine, The ACM's First Electronic Publication, 2001.

[14] **Chen C.,** ***"On the Selection of Image Compression Algorithms"***, Proceedings on Pattern Recognition, Vol. 2, pp. 1500-1504,1998.

[15] **Schneier B.,** ***"Applied Cryptography, Second Edition: Protocols, Algorithms and Source Code in C"***, John Wiley & Sons, Inc., USA, 1996.

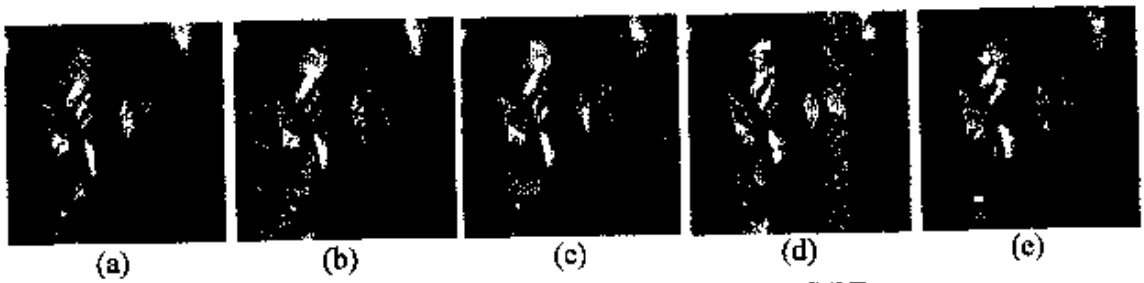
[16] **Stallings W.,** ***"Cryptography and Network Security, Principles and Practice"***, Third Edition, Pearson Education International, Inc., USA, 2003.

[17] **National Institute of Standards and Technology,** FIPS-197-Advanced Encryption Standard (AES), November 2001.

[18] **Beegan A. P.,** ***"Wavelet-based Image Compression Using Human Visual System Models"*** M.Sc. Thesis, Electrical Engineering Department, Virginia

Polytechnic Institute and State University, Blacksburg, Virginia, May 2001.

[19] **Salomon D.,** ***"Data Compression, The Complete Reference",*** Springer-Verlag, Inc., New York, 1998.



(a) (b) (c) (d) (e)

Figure (1): Results of experiment 1 using SPIHT-AES-PE.
(a) Original birds image
(b) Reconstructed image at CR = 1, PSNR = 37.6854 dB
(c) Reconstructed image at CR = 0.5, PSNR = 32.7619 dB
(d) Reconstructed image at CR = 0.25, PSNR = 29.0684 dB
(e) Reconstructed image at CR = 0.125, PSNR = 26.3449 dB

(a) (b)

Figure (2): Results of experiment 2 using EZW-Permutation-PE.
(a) Original birds image.
(b) Reconstructed image.

(a) (b)

Figure (3): Results of experiment 3 using JPEG-Permutation-AES-PE.
(b) Original birds image.
(b) Reconstructed image.

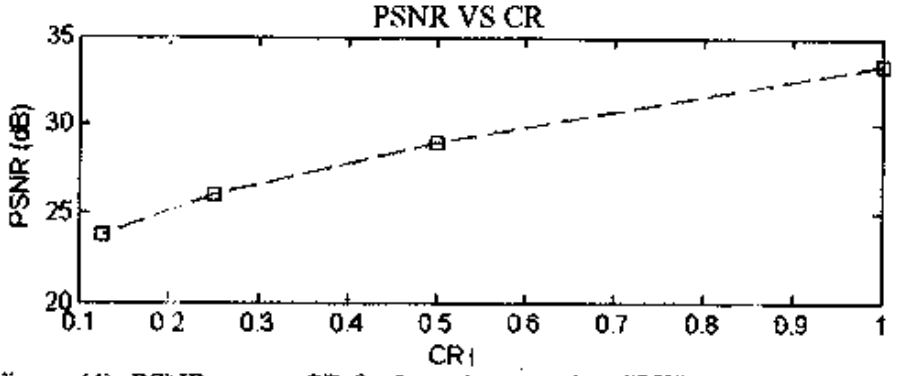


Figure (4): PSNR versus CR for Lena image using SPIHT-AES-PE

Table (1): Experimental results for different CRs of images using SPIHT-AES -PE.

CR	PSNR (dB)			
	Lena	House	Birds	Boys
1	33.4050	35.4861	37.6854	35.4869
0.5	28.9807	30.6352	32.7619	31.4834
0.25	26.0281	26.8452	29.0684	28.5351
0.125	23.7472	24.0476	26.3449	26.2911

Table (2): Experiment 2 results for images using EZW-Permutation-

Image	CR	PSNR (dB)
Lena	0.1439	37.6858
House	0.1700	38.3512
Birds	0.1771	39.7678
Boys	0.1628	37.1407
Average	0.1635	38.2364

Table (3): Experiment 3 results for images using JPEG-Permutation-AES-PE.

Image	CR	PSNR (dB)
Lena	0.3858	34.1462
House	0.3955	33.9120
Birds	0.3181	35.0045
Boys	0.3318	35.1485
Average	0.3578	34.5528