Securing Multimedia Transmission Using Optimized Multiple Huffman Tables Technique

Shaimaa A. El-said, Khalid F. A. Hussein, and Mohamed M. Fouad
Electronics and Communication Department-Faculty of Engineering- Zagazig University- Egypt, Microwaves Department- Electronics research institute- Egypt, Electronics and Communication Department- Faculty of Engineering- Zagazig University- Egypt
Eng.sahmed@windowslive.com, khalid_elgabaly@yahoo.com, fouadzu@hotmail.com

Abstract

Multimedia is one of the most popular data shared in the Web, and the protection of it via encryption techniques is of vast interest. In this paper, an Optimized Multiple Huffman Tables (OMHT) technique is proposed to face some compression and security problems found in Multiple Huffman Tables (MHT) technique. OMHT depends on using statistical-model-based compression method to generate different tables from the same data type of images or videos to be encrypted leading to increase compression efficiency and security of the used tables. A systematic study on how to strategically integrate different atomic operations to build a multimedia compression-encryption system is presented. The resulting system can provide superior performance over both generic encryption and its simple adaptation to multimedia in terms of a joint consideration of security, and bitrate overhead. The effectiveness of this scheme is verified through a series of experiments, and the robustness of our approach is demonstrated by comparing it against a standard compression technique, JPEG on which the MHT technique is built.

Keywords: Multimedia encryption, Multiple Huffman tables, Optimized Multiple Huffman Tables Technique, JPEG, Cryptanalysis.

1. Introduction

With the rapid development of multimedia and network technologies, the security of multimedia becomes more and more important, since multimedia data are transmitted over open networks more and more frequently. Typically, reliable security is necessary to content protection of digital images and videos. To deal with the technical challenges, the major multimedia security technology is Multimedia encryption to provide end-to-end security when distributing digital content over a variety of distributions systems. The security is often achieved by using the naïve (traditional) approach to completely encrypt the entire image, traditional encryption, with a standard cipher [1] (DES, AES, IDEA, etc.).

There are number of applications for which the naive based encryption and decryption represents a major bottleneck in communication and processing. Some recent works explored a new way of securing the content, named, *partial encryption or selective encryption*, by applying encryption to a subset of a bitstream. The main goal of selective

encryption is to reduce the amount of data to be encrypted while achieving a required level of security [2].

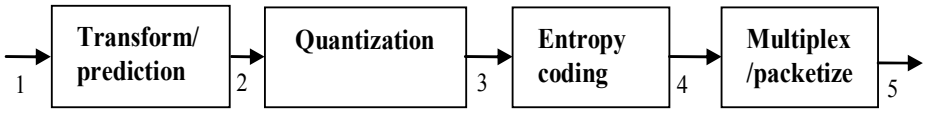


Figure 1. Candidate domains used to apply encryption to multimedia.

According to Fig. 1, there are two straight forward places to apply generic encryption to multimedia. The first possibility is to encrypt multimedia samples before any compression, stages 1 and 2, [3] [4] are examples of ***pre-compression selective encryption***. The main problem with this approach is that the encryption often significantly changes the statistical characteristics of the original multimedia source, resulting in much reduced compressibility. The wavelet-based compression algorithm SPIHT [5], is an example of ***post-compression encryption scheme***, stage 4 and 5. Wu et al proposed encryption scheme based on encoding with multiple Huffman tables (MHT) used alternately in a secret order [6]; is an example of ***in-compression*** selective encryption stage 3. The encryption with reasonably high level of security and unaffected compression can be achieved simultaneously in MHT technique, requiring almost negligible additional overhead. One of the major advantages by using this kind of joint encryption-compression approach is that encryption and compression can be achieved in one single step, which simplifies the system design and makes it flexible for some advanced multimedia processing such as scalability and rate shaping.

The MHT algorithms [6]-[9], aiming to increase the model space while maintaining the computational efficiency, keep the structure of the Huffman tree but enlarge the model space through tree mutation. MHT coding [6] makes use of standard coding tables. It is included in the final bit-stream for every image to be compressed. This approach has some disadvantages:

Visual degradation: very high-visual degradation can be achieved at low bitrate.

Cryptographic security: Gillman and Rivest [10] showed that decoding a Huffman coded bitstream without any knowledge about the Huffman coding tables would be very difficult. However, the basic MHT is vulnerable to known and chosen plaintext attacks as pointed out in [11].

1. It writes all codes of the corresponding tables in the final bitstream even if only some of they were used to encode the associated events of the particular input image.
2. It is not adaptive technique. It does not make use of any statistic about the distribution of the events of the image.

To improve the security several kinds of enhanced MHT schemes have been proposed:

- By inserting random bit in the encrypted bit stream or integrating with a stream cipher [8].
- Recently another scheme via random rotation in partitioned bit streams has been reported [9].

Using known fixed tables in MHT technique generated by mutation (a method introduced in [6]) for compressing and encrypting images causes degradation in both compression ratio and security. We focus our research attention to enhancing multiple

Huffman tables coding techniques. To overcome the drawbacks of MHT technique, a new scheme for more general and efficient secure multimedia transmission, OMHT, is proposed. OMHT depends on using statistical-model-based compression method to generate different tables from a training set has the same data type as images or videos to be encrypted leading to increase compression efficiency and security of the used tables. Performance analysis of the newly proposed scheme OMHT shows that it can provide superior performance over both generic encryption and MHT in the security and compression.

This paper is organized as follows: Optimized Multiple Huffman tables coding technique (OMHT) is described in section 2. Section 3 presents a performance analysis of the newly proposed OMHT technique. Conclusion is given in section 4.

2. Optimized Multiple Huffman Tables Technique

OMHT compression-encryption technique is a modification to the MHT scheme. OMHT generates different Huffman tables for each type of images instead of using fixed Huffman tables for all images as in MHT technique. OMHT compresses an image by converting it into a vector then applies this vector to discreet cosine transformer (DCT). From the output vector only the DCT's coefficients that carry a specific percentage of the image energy are transmitted and the other are neglected, this provides symbols reduction. Then, the selected coefficients undergo the inverse discreet cosine transform (IDCT). The IDCT output is further reduced using a novel quantization technique that depends on dividing the coefficients histogram into a number of vertical slices that have equal areas. The slices number equal to the desired number of quantization levels (QL). The resultant compression ratio and signal to noise ratio varies depending on the chosen number of quantization levels and the amount of transmitted coefficients. The quantizer output is then encrypted by using optimized multiple Huffman tables that is prepared simultaneously with the previous discussed compression procedures. In order to recover the original image at the receiver, the previous stages are processed in the reverse direction as in Fig. 2.

The main advantage of using OMHT technique over other lossy compression technique is that it produces a much smaller compressed file than many compression methods, while still meeting the advantage of encryption without the need to excess time.

As shown in Fig. 2 OMHT process takes two parallel paths A, and B, so it takes no additional time to add encryption to the compressed bitstream as both traditional and selective encryption techniques. Multiple tables are generated for each one of images types: A training dataset is constructed up as a number of selected images of the same characteristics as those to be encrypted. Image types are divided into datasets each dataset contains one N images. Random K images are used to form single Huffman table. Concatenate all pixels values of all ten images in single vector. Then, calculate the occurrence probability of each value, draw the Huffman tree, then generate Huffman table. Now we obtain D different (not fixed) Huffman tables for each dataset. The number of tables that can be obtained from N images is:

$$D = C_{K}^{N}$$

A. Image Compression using OMHT

Following is the procedure of compressing the original image as shown in Fig. 2 path A.

1. The input nXm image is converted into a single vector by cascading rows to form single vector.

2. This vector is exposed to a discrete cosine transformer to transform the spatial domain of an image into its frequency domain in which the image energy concentrated in small number of coefficients.

3. The output of the DCT process is a vector that have the same length of the image (number of pixels in the image), but with many values approximated to zeros. A selector is used to select the number of transmitted coefficients (*Tc*) that contains a specific percentage of image energy where as the other coefficients are canceled.

4. The selected coefficients are returned back into spatial domain using Inverse Discrete Cosine transformer.

5. The IDCT output is quantized. Quantization technique is irreversible; this means that the dequantized values can't be turned back to their original values leading to quantization losses. The quantization procedures are as follows:

- The area under the histogram of the IDCT output is divided into a number of vertical slices with equal area. The number of these slices is equal to that of quantization levels.
- Each slice has start and end points. the midpoint value of each slice range is considered as a quantization level.

All coefficients lie in a slice range take the same value as that of slice midpoint, and hence the transmitted values are reduced.

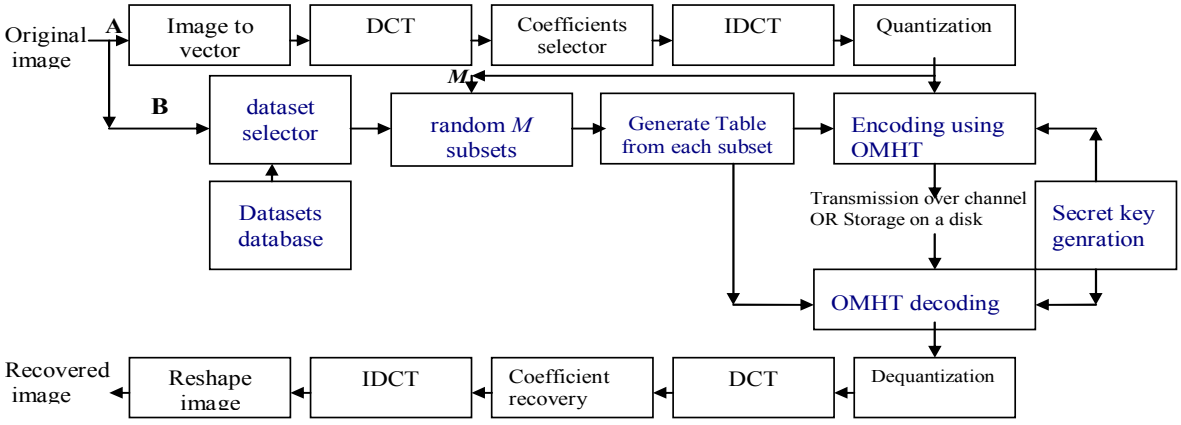


Figure 2. Optimized multiple Huffman table (OMHT) coding system

B. *Preparing And Using The Optimized Tables*

Following is the procedure of preparing and using the optimized multiple Huffman tables and how it is used to both encode and encrypt images as shown in Fig. 2 path B.

Step 1: images training set are divided into datasets. Each dataset's images have the same properties.

Step 2: each dataset contains one *N* image.

Step3: The input image compared to datasets to select the dataset that has the same properties.

Step4: randomly choose *M* subsets each subset contains *K* images from the dataset. Concatenate all images of each subset and calculates the pixels probabilities. Then

draw Huffman tree and find the Huffman table contains the different pixels' values and their associated variable codewords. Now we have *M* different tables to be used.

Step5: Tables are saved at each decoder, and the order by which the tables are generated and used is kept secret.

Step6: Number the generated *M* tables from 0 to *M*-1.

Step7: Generate a random vector P (the secret order) its length equal to the length of image under consideration. Each element value in P ranges from 0 to *M*-1.

Step8: For the i^{th} encountered symbol (coefficient to be encoded), use table $p_{i \,(\mathrm{mod}\ n)}$ to encode it.

3. OMHT Performance Analysis

In this section the performance of the proposed technique is analyzed from the viewpoints of compression and encryption. Experiments based on testing OMHT coding technique to both encode and encrypt a gray-level test images. The training set was obtained from nine different 512×512 images. All the images are 256 grey scale images.

Security analysis of the encryption algorithm is commonly needed for evaluating and comparing the performance of encryption algorithms. Performance analysis based on cryptanalysis can prove the complexity for the attacker deciphering the encryption algorithm in theory, but cannot provide the visual security degree of the cipher-images. In order to develop an objective assessment algorithm on visual security degree of visual media, current security assessments methods of image and video encryption were deep studied and divided into three kinds: assessment based on cryptographic analysis, assessment based on subjective evaluation, and assessment based on video quality assessment.

3.1. Assessment Based On Video Quality

Video quality can be evaluated by measuring compression ratio, peak signal to noise ratio, mean square error, and signal to noise ratio of compressed images.

Experiment 1 uses lossy OMHT to encrypt and compress the test images using histogram equal areas division quantizing technique. It gives the ability to control the compression ratio and peak signal to noise ratio by either change QL while the amount Tc is constant, or changing the amount of Tc while QL is constant.

(a) Original Image (b) Decoded for T_C=99.9% (c) Decoded for T_C= 99.6% (d) Decoded for T_C=99.4%

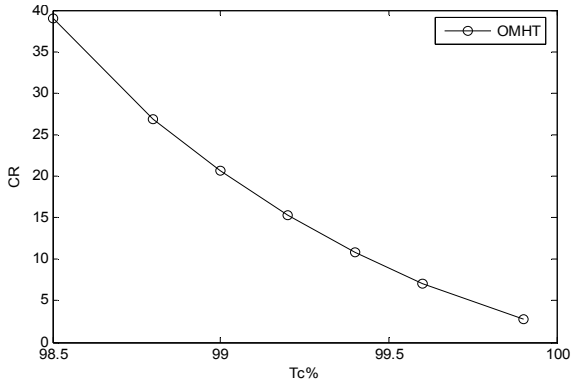
(e) Decoded for T_C=99.2% (f) Decoded for T_C= 99% (g) Decoded for T_C= 98.8% (h) Decoded for T_C=98.5%

Figure 3. The effect of reducing number of transmitted coefficients on image visual degradation

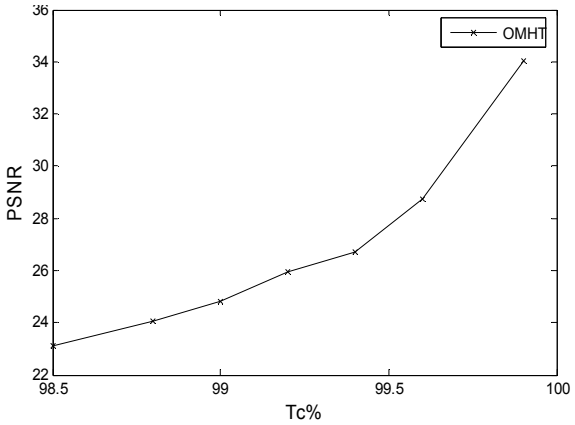
As shown in Table 1, Fig. 3, and Fig. 4 while the number of quantization levels is constant at q=128 and the amount of transmitted DCT's coefficients changes from Tc=98.5% of the image energy to Tc= 99.9%. As Tc increases, the CR decreases providing an increase in PSNR and in the image quality.

Table 1. Compression performance of applying OMHT with different transmitted coefficients number

TC=	99.9%	99.6%	99.4%	99.2%	99%	98.8%	98.5%
CR	2.82	7.052	10.86	15.252	20.61	26.94	38.97
BPP	2.84	1.135	0.737	0.5245	0.388	0.297	0.207
PSNR	34.03	28.74	26.71	25.929	24.79	24.07	23.13
MSE	20.85	94.22	116.7	179.29	200.8	290	360
SNR	26.95	21.67	19.62	18.845	17.72	16.98	16.04



(a) Variation CR with Varying OMHT Number of *Tc*



(b) Variation of PSNR with Varying of *Tc*

Figure 4. The effect of reducing OMHT number of *Tc* on CR, and PSNR

Table 2. Compression performance of applying OMHT with varying the number of quantization levels

	Q=4	Q=8	Q=16	Q=32	Q=64	Q=128	Q=256
CR	24.08	18.437	15.99	13.25	11.45	10.49	10.14
BPP	0.3322	0.4339	0.500	0.604	0.6987	0.762	0.7889
PSNR	22.09	23.295	24.42	26.14	27.59	27.74	27.92
MSE	382.2	305	222.21	119.67	102	98.12	80.94
SNR	15.00	16.21	17.33	19.06	20.50	20.66	20.84

As shown in Table 2, and Fig. 5, while the amount of transmitted coefficients is constant at Tc=99.5% and the number of quantization levels changes from using four quantization levels to using 256 quantization levels, the compression ratio decreases providing an increase in peak signal to noise ratio.

Experiment 2 uses lossy JPEG standard compression technique [12]-[15] to compress the Lena test image with varying the number of quantization levels.

As shown in Table 3, Fig. 6, and Fig. 7, as long as the values in the Q matrix becomes larger and larger the image compressed more and more leading to decrease in peak signal to noise ratio and the appearance of visual degradation (blockness effect) because of the significant block artifacts. Fig. (7-h) shows a decoded image of low bit rate image (0.2bpp) compressed by jpeg. It is unacceptable in certain applications.

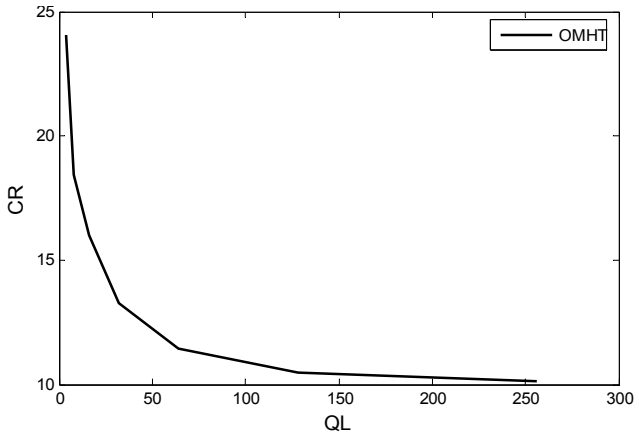
Table 3. Compression performance of applying JPEG with different quantization degree on lena image

q=	Q/2	Q	2Q	4Q	6Q	8Q	16Q
CR	4.25	7.0427	10.317	15.21	20.32	24.7	39.80

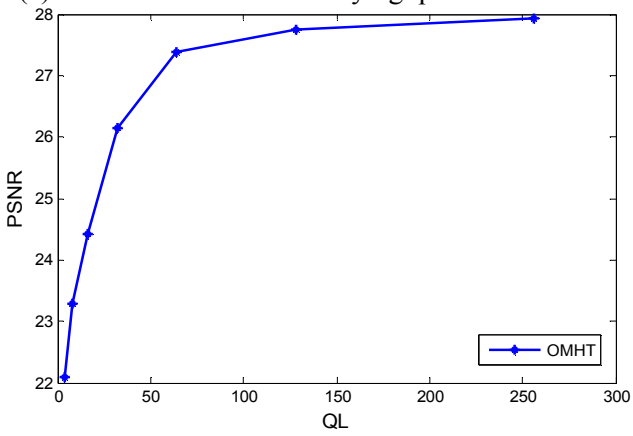
BPP	1.88	1.136	0.7754	0.526	0.394	0.324	0.201
PSNR	41.72	30.05	27.76	25.69	24.42	23.5	21.15
MSE	4.179	61.29	103.85	167.25	224.2	276	476
SNR	34.63	22.97	20.675	18.61	17.33	16.42	14.06

Table 4.Comparison of CR between OMHT, and JPEG at different BPP on Lena Image

BPP	OMHT	JPEG
0.2	39	39.01
0.18	44.4	43
0.16	49.4	49
0.14	57.4	58.02
0.12	65.6	65.3
0.1	73.9	73



(a) Variation of CR with varying quantization levels

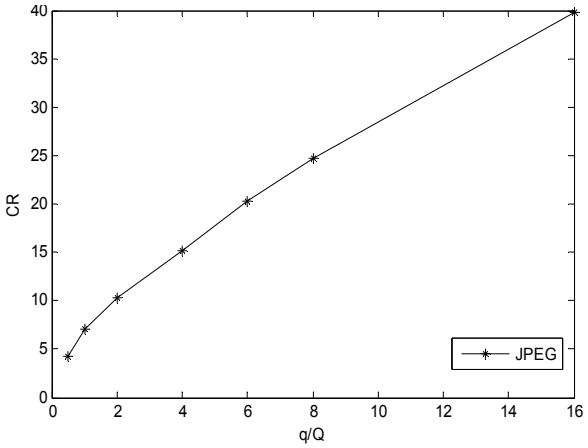


(b) Variation of PSNR with varying quantization levels

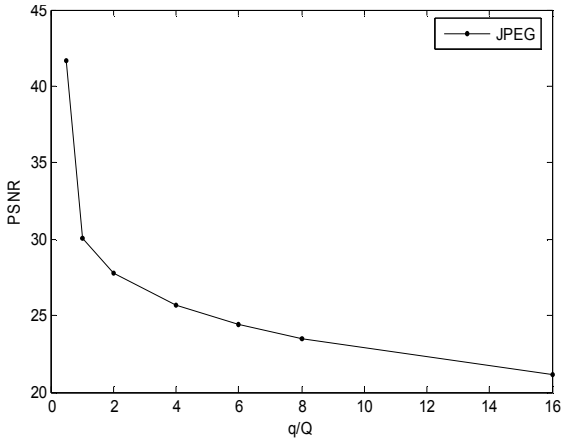
Figure 5. The effect of reducing OMHT number of quantization levels on CR, and PSNR

Table 5.Comparison of PSNR between OMHT, and JPEG at different BPP on Lena Image

BPP	OMHT	JPEG
0.2	22.6	21.14
0.18	22.2	20
0.16	21.9	19.4
0.14	21.6	18
0.12	21.3	16.7
0.1	21	15

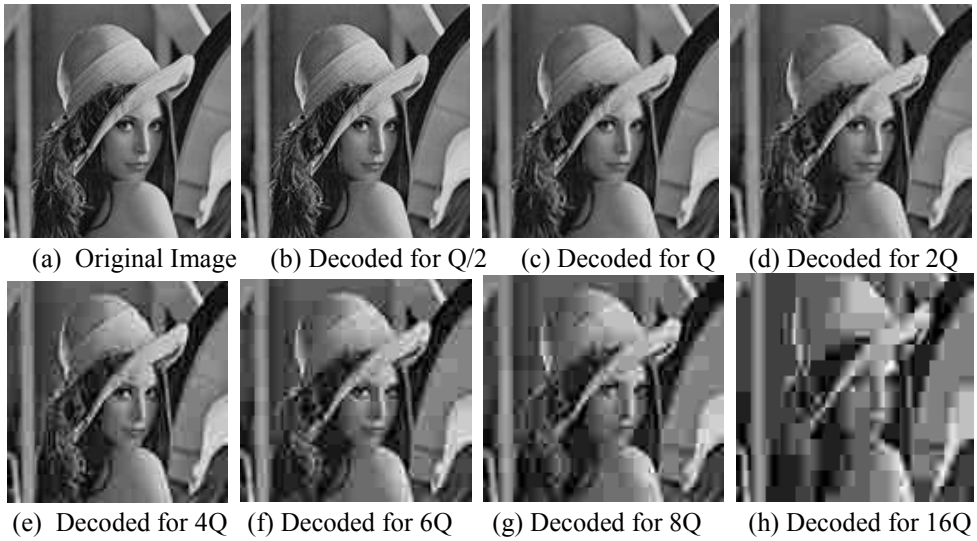


(a) Variation of CR with varying JPEG quantization levels



(b) Variation of PSNR with varying quantization levels

Figure 6. The effect of reducing JPEG quantization levels on CR, and PSNR



(a) Original Image (b) Decoded for Q/2 (c) Decoded for Q (d) Decoded for 2Q

(e) Decoded for 4Q (f) Decoded for 6Q (g) Decoded for 8Q (h) Decoded for 16Q

Figure 7.The effect of reducing JPEG quantization levels on Lena image visual degradation

Experiment 3 The compression performance of lossy OMHT is compared to that of lossy JPEG technique. From Fig. 8 shows that OMHT behaves better than JPEG at low bitrate in CR, PSNR, and image appearance. From Tables 4, 5, and 6, it is obvious that lossy OMHT technique not only efficient in encryption than using multiple Huffman tables (MHT) technique but also more effective in storage space and transmission bandwidth required than both MHT and JPEG especially at low bitrates.

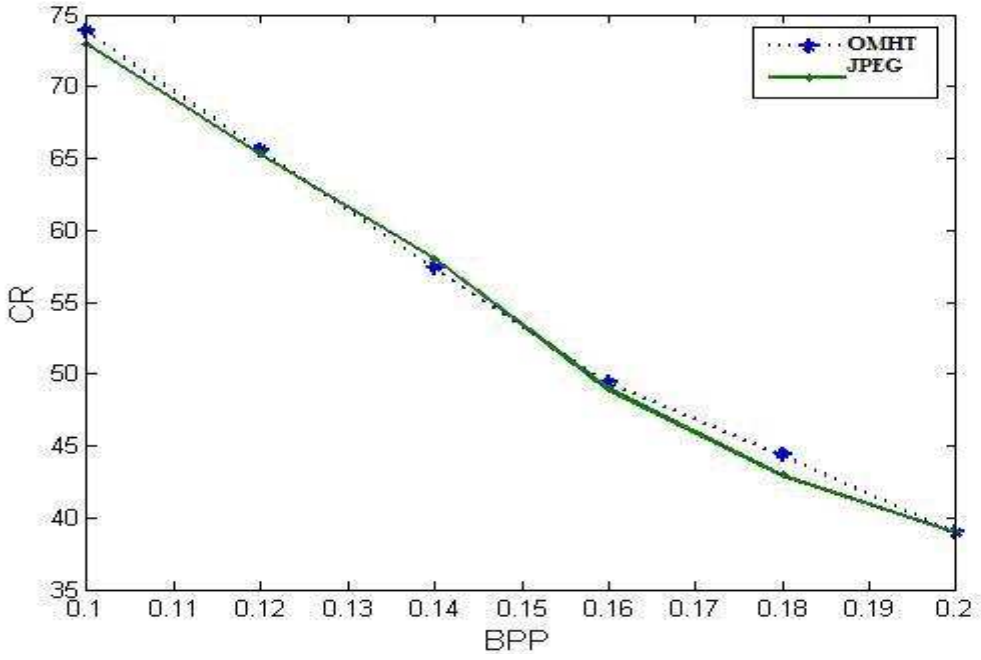


Figure 8. CR of both OMHT technique and JPEG for Lena image

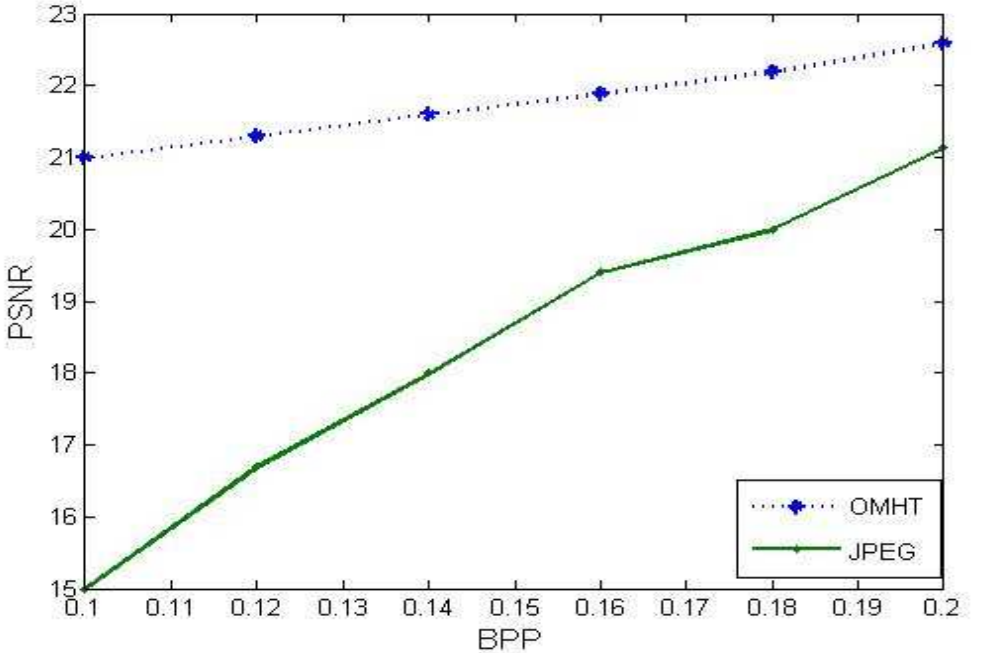


Figure 9. PSNR of both OMHT and JPEG for Lena image

Table 6. Comparison between OMHT, and JPEG compression performance on the nine test images

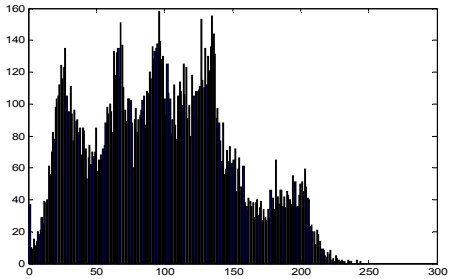
	Bridge		Pirate		Camera man		Pilot		Lena		Living-room		Mandril_gray		Woman_blond		Woman-darkhair		average	
	OMHT	JPEG	OMHT	JPEG	OMHT	JPEG	OMHT	JPEG	OMHT	JPEG	OMHT	JPEG	OMHT	JPEG	OMHT	JPEG	OMHT	JPEG	OMHT	JPEG
CR	13.327	12.23	65.89	23.91	13.108	19.143	26.08	17.38	13.26	9.64	59.98	22.67	30.76	15.99	164.78	60.302	620.09	70.53	111.9	27.98
BPP	0.600	0.653	0.121	0.334	0.610	0.417	0.306	0.460	0.50	1.13	0.13	0.35	0.2601	0.50	0.0485	0.1327	0.0129	0.1134	0.288	0.454
PSNR	23.941	23.94	26.10	28.51	25.829	25.786	24.46	26.58	26.14	27.05	26.10	28.79	24.687	26.77	23.2219	21.855	26.397	26.24	25.29	27.16
MSE	330.3	262.2	179.2	81.74	213.88	171.58	292.6	142.9	189.6	140.2	200.6	85.85	218.48	107	244.63	266.14	187.63	154.54	245.76	156.9
SNR	20.17	20.18	20.14	22.55	20.098	20.055	20.07	22.19	20.25	20.41	20.17	22.85	20.18	22.26	20.1345	18.768	20.150	19.99	20.2	21.01

Table 6 shows compression performance comparison between using OMHT coding technique in encrypting and compressing the test images and using JPEG coding technique which provides only compression on all the nine test images.

3.2. Assessment Based On Subjective Evaluation

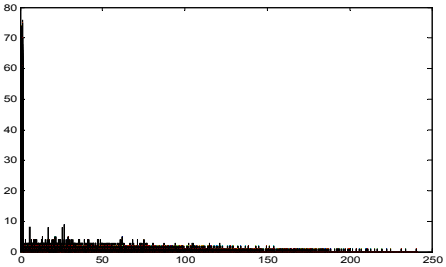
Experiment 4 measures the unrecognizable degree of cipher images. It measures the encryption strength performance of the proposed OMHT technique, Lena image in Fig. (10-a) and Fig.(10-b) is compressed and encrypted using the OMHT uses multiple Huffman tables, generated from a large set of training images that have the same type of the test image not a predetermined fixed tables, in a secret order. Fig. (10-c), and Fig. (10-d) show the test image and its histogram after decoding it with another technique as JPEG. Fig. (10-e), and Fig. (10-f) show the test image and its histogram after decoding it with OMHT technique and the same encoding tables but without knowing the secret order (secret key).

(a) Original Image

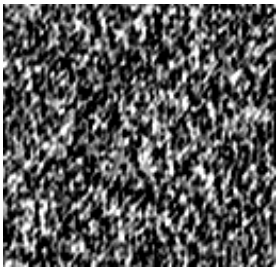


(b) Histogram of Original Image

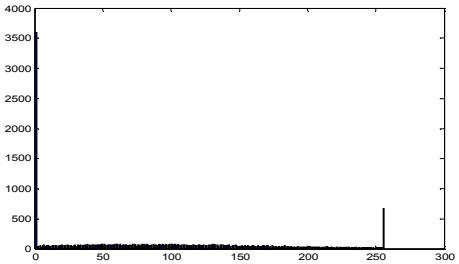
(c) Decoded image with JPEG



(d) Histogram of decoded Image by JPEG



(e) Decoded image with OMHT



(f) Histogram of decoded Image by OMHT

Figure 10. The effect of decoding Lena image without the secret order

3.3. Assessment based on cryptographic analysis

Cryptanalyst analyzes the possibility of deciphering the cipher images through the use of cryptanalysis attack. To evaluate the security of the proposed scheme it is important to consider how cryptanalysts would attack the cryptosystem. Attack models are usually classified into three categories. In each model, every detail of the encryption/decryption algorithm is assumed to be known to cryptanalysts. The security depends solely on the hidden key.

- *Ciphertext-only attack*

The cryptanalyst has only the ciphertext to work with; brut-force exhaustive key search attack. The strength of resisting this attack relies on a larger key space. As described previously the available number of tables (D) that can be generated from N images in the selected dataset by using random K subsets for generating a single table is calculated as:

$$D = C_{K}^{N}$$

Since the length of the random vector by which the tables are used is equal to n; the length of the compressed vector and each element ranging from 0 to M-1, where M is the randomly generated tables. There would be (M)n different orders to use the Huffman tables. As a result the size of the key space is:

$$size(keyspace) = C_{M}^{D} \times (M)^{n}$$

Let each dataset contains one thousand images of the same type. If random ten images are selected to form single Huffman table, we can obtain about 3762*1020 different tables. If there is 7000 coefficients (min. number of coefficients after compressing the image values), we need to generate a random vector that contains 7000 elements each varying from 0 to M. The resulting size of key space would at least be:

$$size(keyspace) = C_{7000}^{3762\times 10^{20}} \times (7000)^{7000}$$

So, it is practically impossible to perform brute-force search in a key-space of this size.

- *Known-plaintext attack*

  In this attack model, the cryptanalyst has a string of symbols and its corresponding

encrypted bitstream. The goal is to obtain which tables are used and in which order. The first step is to guess which bits in the bitstream correspond to which symbol in the original data. Since symbols are encoded with different Huffman tables would produce codewords with slightly different lengths, synchronization between plaintext and ciphertext is extremely difficult.

- *Chosen-plaintext attack*
  Cryptanalyst could chose the plaintext and obtain the corresponding ciphertext. His goal is to obtain the secret key to decrypt the ciphertext in the future. In most encryption schemes the secret key remains unchanged for along time. These techniques are vulnerable to this kind of attack. The attacker inserts one single symbol into the cipher and observes the corresponding output codeword. In this case the attacker would have no synchronization problem at all. In OMHT, the secret key is the order by which we choose the Huffman tables and using it. OMHT technique is designed to receive symbols as a whole chunk and output the corresponding codewords all together. To increase the security the chunk size should be as big as possible.

4. Conclusion

From the viewpoint of compression, the experiments' results reveal that the proposed OMHT technique achieves better compression and security performance than both MHT, and JPEG Image Compression Standard. The experiments on the proposed technique and JPEG also reveal that the resultant CR of lossy OMHT technique can be increased over a wide range for the same image according to the number of quantization levels chosen with a slightly decrease of its PSNR without a great noticeable visual degradation at low bitrate, where as increasing compression ratio of both MHT, and JPEG techniques is combined by the appearance of the blockings effect (visual degradation) in the reconstructed image at low bitrate, this enables the usage of lossy OMHT technique not only in applications require high security but also in many applications need a compression ratio variability with nearly stable PSNR. Whereas, the proposed joint compression-encryption (OMHT) technique employ a Source Symbols Reduction using one dimensional DCT, quantization, and Huffman Coding enhance the performance of the compression. Further, the proposed new compression-encryption technique could be applied on any source data, not only images, which uses Huffman coding. OMHT is a general technique; it is suitable for compression and encryption of text, image, and video files.

From the viewpoint of security, the experiments' results reveal that the proposed OMHT technique achieves better security than MHT technique since it is more resistible to ciphertext-only attack, known-plaintext attack, and even chosen-plaintext attack.

- Joint compression-encryption OMHT technique achieves both high security and compression performance in one single step, which simplifies the system design and reduces time required to perform compression followed by encryption.
- Since images have different statistics, using the same fixed JPEG standard predefined coding tables as suggested in MHT technique will not be effective in encoding all image and video types.
- The OMHT method obtains better performance in terms of storage space use and more stable peak signal to noise ratio than that of JPEG in encoding an image with small and great gray-level variations among adjacent pixels.
- Receivers haven't the secret order cannot decode the encoded images successfully.

References

[1.] W. Stallings, Cryptography and Network Security Principles and Practices. Upper Saddle River, NJ: Prentice Hall, 2003.
[2.] M. Van Droogenbroeck and R. Benedett, "Techniques for a selective encryption of uncompressed and compressed images," in Proceedings of Advanced Concepts for Intelligent Vision Systems (ACIVS '02), pp. 90–97, Ghent, Belgium, September 2002.
[3.] L. Qiao, K. Nahrstedt, and M.-C. Tam, "Is MPEG encryption by using random list instead of zigzag order secure?," in Proceedings of the IEEE International Symposium on Consumer Electronics (ISCE '97), pp. 226–229, Singapore, December 1997.
[4.] T. Uehara and R. Safavi-Naini, "Chosen DCT coefficients attack on MPEG encryption scheme," in Proceedings of IEEE Pacific Rim Conference on Multimedia, pp. 316–319, Sydney, Australia, December 2000.
[5.] H. Cheng and X. Li, "Partial encryption of compressed images and videos," IEEE Transactions on Signal Processing, vol. 48, no. 8, pp. 2439–2451, 2000.
[6.] C.-P. Wu and C.-C. J. K. Kuo, "Design of integrated multimedia compression and encryption systems," IEEE Transactions in Multimedia, vol. 7, no. 5, pp. 828–839, 2005.
[7.] C.-P. Wu and C.-C. Kuo, "Efficient multimedia encryption via entropy codec design," Proc. SPIE, vol. 4314, Jan. 2001.
[8.] Xie and C. J. Kuo, "Enhanced Multiple Huffman Table (MHT) Encryption Scheme Using Key Hoping," In Proceedings of IEEE International Symposium on Circuits and Systems, pp.568–571, May2004.
[9.] Xie and C. J. Kuo ,"Multimedia Data Encryption via Random Rotation in Partitioned Bit Stream," In Proceedings of IEEE International Symposium on Circuits and Systems, pp.568–571, May2004.
[10.] W. Gillman and R. L. Rivest, "On breaking a Huffman code," IEEE Transactions on Information Theory, vol. 42, no. 3, pp. 972–976, 1996.
[11.] J. Zhou, Z. Liang, Y. Chen, and O. C. Au, "Security analysis of multimedia encryption schemes based on multiple Huffman table," IEEE Signal Processing Letters, vol. 14, no. 3, pp. 201–204, 2007.
[12.] W. Pennebaker and J. Mitchell, JPEG Still Image Data Compression Standard. Van Nostrand Reinhold, New York, 1993.
[13.] http://www.jpeg.org (JPEG resources)
[14.] http://www.jpeg.org/public/jfif.pdf (JPEG file interchange format)
[15.] (independent JPEG group) ftp.uu.net:/graphics/jpeg