

Coding Implementation By Using Discrete Wavelet Transform

Dr.Matheel Emad AL-Deen AL-Dargazli* &
Dr.Nuha Abduljabbar Rajab*

Received on:1/6/2009

Accepted on:5/11/2009

Abstract

Image processing can be considered as an essential part of wide range computer applications. This application deals with image coding and it discusses image coding schemes, by using transforms to code the image or not. Huffman coding and Run Length coding are not using transforms to code the image, moreover, threshold coding and Zonal coding use transforms to code the image. Two types of transforms, Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT) have been used after segmenting the image into blocks, and then Zonal coding algorithm or threshold coding algorithm is used to get the coded image. Finally we conclude that using wavelet transform is better than other transforms when it is used for image coding.

ترميز الصور باستخدام التحويل المويجي

الخلاصة

تعتبر معالجة الصور جزءاً أساسياً لمدى واسع من تطبيقات الحاسبات. هذا التطبيق يعني بترميز الصورة وناقش أساليب ترميز الصورة باستخدام التحويلات لترميز الصورة أو لا. ترميز هافمان وطول التنفيذ لم يستخدم التحويل لترميز الصورة. بالإضافة لذلك ترميز حد العتبة وترميز زونال استخدم التحويلات لترميز الصورة. هناك نوعين للتحويلات، التحويل باستخدام الجيب تمام المنقطع والتحويل باستخدام الموجة المنقطع والذي استخدم بعد تقطيع الصورة الى مقاطع. بعد ذلك تستخدم خوارزمية زونال للترميز أو خوارزمية حد العتبة للترميز للحصول على الصورة المرمزة. وأنتجت في نهاية البحث استخدام النقل المويجي كان أفضل من باقي أنواع النقل المستخدم لترميز الصور.

1 Introduction

The attention of the image processing community was soon caught when Daubechies and Mallat in addition to others, contributed to the theory of wavelets by establishing connections to discrete signal processing results. Since there are numbers of theoretical as well as practical contributors made, various aspects of the wavelet transform and the subject have grown rapidly [1].

Wavelet theory covers quite a wide area, it treats both continuous and the discrete time cases. It provides very general techniques that can be applied to many tasks in signal processing, and therefore has numerous potential applications. In practice, the wavelet transform is of interest in non stationary signals because it provides an alternative to the classical Short Time Fourier Transform (STFT) or what is known as (Gabor transform), because the Wavelet Transform (WT) relates to time frequency analysis [2,3].

Functionally, Discrete Wavelet Transform (DWT) is very much similar to the discrete Fourier transform, in that the transformation function is orthogonal.

The wavelet can be regarded as the most efficient transform that deals with image, sound, or any other pattern since it provides a powerful time-space (Time-Frequency) representation [4].

The effect of the wavelet transform on the image is defined as "it abstracts the image in universal space, this contains all information into an approximately orthogonal (statistical independent) spaces (represent projection of the subject) contained in this universal space. By changing the scale in each projection, the algebraic sum of all these projection gives the original subject before abstraction process [5, 6].

2 Coding Classification

Coding schemes are classified as coding without using transform (lossless) and coding using transform (lossy) types. Schemes like the Huffman coding, run-length coding, arithmetic coding, predictive coding and bit plane coding belong to the lossless class [7]. Coding with transform methods may be constructed through a variety of ways. The lossy methods include predictive coding, transform coding and the combination of both which is called hybrid coding.

3 Image Coding Without Using Transforms

This type of coding involves a completely reversible scheme by which the original data can be reconstructed exactly. An important concept here is the idea of measuring the average information in an image, referred to as the Entropy.

The Entropy for an $N \times N$ image can be calculated as in bit per pixel:

$$\text{Entropy} = - \sum_{i=0}^{L-1} P_i \log_2 P_i \quad \dots (1)$$

where $P_i = n_k / N^2$.
 n_k = the total number of pixels with gray value k .

L = the total number of gray level (e.g, 25, for 8 bit).

This measure provides us with a theoretical minimum for the average number of bits per pixel that could be used to encode the image. This number is theoretically optimal and can be used as a metric for judging the success of a coding scheme [8].

In general, this image coding scheme deals with the value or brightness of each individual pixel and guarantees that the value of each pixel in the reconstructed image will match its corresponding original value. Run-length coding and

Huffman coding are some of the coding methods and they are used in image compression [9].

3.1 Huffman Coding

The Huffman code, developed by D. Huffman in 1952, is a minimum length code. This means that given the statistical distribution of the gray levels (the histogram), the Huffman algorithm will generate a code that is as close as possible to the minimum bound, the entropy. This method results in a variable length code, where the code words are of unequal length [10].

The Huffman algorithm can be described in five steps:

1. Find the gray-level probabilities for the image (by finding the histogram).
2. Order the input probabilities (histogram magnitudes) from smallest to largest.
3. Combine the smallest two by addition.
4. Go to step 2, until only two probabilities are left.
5. By working backward a long the tree, generate a code by alternating assignment of 0 and 1.

3.2 Run Length Coding (RLC)

It is an image coding method that works by counting the number of adjacent pixels with the same gray-level value. This count, called the run length, is then coded and stored-basis methods that are used primarily for binary (two-valued) images and extended versions for gray-scale images.

Basic RLC is used primarily for binary images but can work with complex images that have been preprocessed by thresholding to reduce the number of gray levels to two [10].

There are three steps to implement basic RLC, these are:

1. Define the required parameters. It can either use horizontal RLC, counting along rows, or vertical RLC, counting along the columns. In Basic horizontal RLC the number of bits used for the coding depends on the number of pixels in a row. If the row has 2^n pixels then the required number of bits is n , so that a run that is the length of the entire row can be coded.
2. The next step is to define a convention for the first RLC number in a row (column) if it represents a run of 0's or 1's.
3. Then coding the image by counting the consecutive 0's and 1's for each row (column) and the first RLC number will be 0 depending on the convention that is defined.

4 Image Coding Using Transform

This coding rather than deal with the value of every pixel, the coding schemes attempt to determine visually important components of an image. By maintaining visually important information, the reconstructed image has a similar look as the original [8].

Transform coding is a very popular method for still image coding and intra-frame or inter-frame error image content and to encode transform coefficients rather than the original pixels of the images [8], transform coding is a form of block coding done in the transform domain [10]. The primary reason is because this transform is effective, it efficiently puts most of the information into relatively few coefficients.

So many of the high-frequency coefficients can be quantized to (0) (eliminated completely).

Two particular types of transform coding have been widely

explored : Zonal and threshold coding. These two types vary in the method they use for selecting the transform coefficients to retain (using ideal filters for transform coding selects the coefficients based on their location in the transform domain) [10].

4.1 Zonal Coding

This coding involves selecting specific transform coefficients based on maximum variance (The square of the standard deviation (STD)) [10]. In Zonal coding:

1. a Zonal mask is determined for the entire image by finding the variance for each frequency component.
2. this variance is calculated by using each subimage within the image as a separate sample and then finding the variance within this group of subimages.

4.2 Threshold Coding

This coding selects the coefficients above a specific value [10], where each transform coefficients is compared with a threshold so,

1. If smaller than threshold, set to zero.
2. If larger than threshold, retain for encoding.
3. Threshold is determined after evaluation of all coefficients.
4. Address of retained coefficients has to be sent to receiver as side information [10]. In threshold coding a different threshold mask is required for each block, which increases file size as well as algorithmic complexity.

Thresholding process has different types as will be given below. The choice of thresholding

method depends on the application. Thresholding operations are applied to the coefficients of the wavelet transforms as seen from the capital letters used in the mathematical notations used below [11]. Hard thresholding is also called “kill / keep” strategy [12] or “gating” [13]. If the signal or coefficients value is below a present value it is set to zero, that is [7]:

$$\hat{X}_k^j = T_n(G_k^j, Thv) = \begin{cases} G_k^j, & |G_k^j| > Thv; \\ 0, & |G_k^j| \leq Thv \end{cases} \dots (2)$$

where
Thv: is the threshold value or the gate value.

\hat{X}_k^j :reconstruced value.

G_k^j :original value.

Soft Thresholding is an alternative scheme of hard thresholding and can be started as:

$$\hat{X}_k^j = T_s(G_k^j, Thv) = \begin{cases} \text{sig}(G_k^j) * (|G_k^j| - Thv), & |G_k^j| > Thv \\ 0, & |G_k^j| \leq Thv \end{cases} \dots(3)$$

Or in another form [13]:

$$\hat{X}_k^j = Ts(G_k^j, Thv) = \text{sig}(G_k^j) * (|G_k^j| - Thv)_+$$

Where

$$\text{sig}(G_k^j) =$$

$$\left\{ \begin{array}{l} +1, \text{ if } G_k^j > 0; \\ 0, \text{ if } G_k^j = 0; \\ -1, \text{ if } G_k^j < 0 \end{array} \right. \quad \text{and } (q)_+ = \begin{array}{l} q, \text{ if } q > 0; \\ 0, \text{ if } q \leq 0 \end{array} \quad \dots (4)$$

Hard thresholding can be described as the usual process of setting to zero the wavelet coefficients whose absolute values are less than or equal to the thresholding value (Thv).

Soft thresholding is an extension of hard thresholding, firstly setting to zero the wavelet coefficients whose absolute value is less than or equal to (Thv), then shrinking the non zero coefficients towards zero by a threshold value (Thv). As can be seen in Equation (2), the hard - thresholding procedure creates discontinuities at $G = \pm \text{Thv}$, while soft-thresholding does not [14]. Hard thresholding, indeed, seems more natural to non-statisticians [15].

5 Experimental work

An image coding scheme consists of two components, an encoder and a decoder. The encoder is the portion of the scheme that takes an image. The decoder performs the opposite actions of the encoder; it takes the encoded image and attempts to reconstruct the original image. Figure (1) shows this process.

This process will either be lossless (with transform) or lossy (with transform), which will be determined by the particular needs of the user.

5.1 Image Coding Algorithms With -out Using Transforms

This type of image coding does not use transforms to code the image, such that, there is no loss of

information where the decoded image exactly looks like the original image because this type of coding is recoverable. Run length coding algorithm and Huffman coding algorithm does not use transform to code the image.

In the run length method, the original image is color, to encode the image we convert it to a binary image, now each image is divided into $T \times V$ distinct blocks, where both T and V are smaller than the image size, and *RLC* algorithm is performed over each block of the $T \times V$ blocks separately to have at last the coded image.

The decoded (reconstructed) operation is performed on each distinct block individually to determine the values of the pixels in the corresponding block of the output (reconstructed or decoded) image [16].

The proposed algorithm of image coding using distinct block processing can be illustrated as follows:

1. Divide the binary image ($N \times N$) into ($T \times V$) size of distinct blocks. Note that, here T and V are selected to be equal to $N / \text{sqrt}(b)$, $b = 2^{2n}$, $n=1,2,3,\dots,N$.
2. Apply the *RLC* algorithm to the first block as mentioned in chapter two to get the coded image.
3. Take the inverse *RLC* to get back our original image.
4. Repeat steps (2-3) until all blocks are completed.
5. The decoded image is the concatenation of all the processed distinct blocks of step (4). The binary fish image was coding by using run length coding algorithm and the results were shown in Figure (2).

- **Huffman Coding Algorithm**

In this method, the original image is color, we code the image after converting it to a grayscale image, then each image is divided into $T \times V$ distinct blocks, where both T and V are smaller than the image size and Huffman algorithm is performed over each block of the $T \times V$ block separately to have our coded image.

The decoded operations are performed on each distinct block individually to determine the values of the pixels in the corresponding block of the decoded image.

The proposed algorithm of image coding using distinct block processing can be illustrated as follows [16]:

1. Divide the grayscale image ($N \times N$) into ($T \times V$) size of distinct blocks. Note that, here T and V are selected to be equal to $N / \text{sqrt}(b)$, $b = 2^{2n}$, $n = 1, 2, 3, \dots, N$ [16].
2. Apply the Huffman algorithm to the first block to get the coded image.
3. Take the inverse Huffman to get back our original image.
4. Repeat steps (2-3) until all blocks are completed.
5. The decoded image is the concatenation of all the processed distinct blocks of step (4).

The Rose image was coding by using Huffman coding algorithm and the results were as shown in Figure (3).

5.2 Image Coding Algorithm with Transforms

This type of image coding using transforms to encode the image. Where the image is divided into blocks or subimages, and the

transform is calculated for each block. After the transform has been calculated, the transform coefficients are coded. Here the decoded image has a similar look to the original image with a small value of Mean Square Error (MSE).

In this method, each image is divided into $T \times V$ distinct blocks, where both T and V are smaller than the image size. The Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT) transforms is performed over each block of the $T \times V$ blocks separately. Then the threshold method is performed over each block.

The inverse threshold is taken for each block and then decoded operation is performed on each distinct block individually to get the output (decoded) image. The proposed algorithm of image decoding using distinct block processing can be illustrated as follows [16]:

1. Divide the ($N \times N$) gray scale image into ($T \times V$) blocks, where each block is transformed and coded separately.
2. Take from the given image of ($N \times N$) the first ($T \times V$) blocks, so that, T and V are selected to be equal to $N / \text{sqrt}(b)$, $b = 2^{2n}$, $n = 1, 2, 3, \dots, N$.
3. Transform each block ($T \times V$), by using these transformation (DCT, DWT).
4. Calculate the threshold value of the block by :

$$\text{thv} = \text{sqrt} (2 * \log (2 * M)) *$$

sig

$$\text{sig} = (\text{norm} (\text{double} (W) - \text{mean} (\text{mean} (W)), 'fro' / M) / \text{SNR} \dots (6)$$

where

M = the dimension of the input matrix

W = the input matrix
 Fro = Frobenius norm condition number [7].

Apply threshold coding algorithm to the transformed block using the thresholding formula:

$$C_T(u,v) = \begin{cases} C(u,v) - Th_v & \text{if } C(u,v) > Th_v \\ 0 & \text{otherwise} \end{cases} \dots(7)$$

5. Apply inverse threshold coding to the coded image block.
6. Take the inverse transform to the block to have the decoded image block.
7. Repeat steps (3-7) until all blocks are completed.
8. The decoded image is the concatenation of all the processed distinct blocks of step 8.

The Tiger image was coded by using threshold coding algorithm and the results are shown in Figure (4) that represents the results of coding the image with DCT and Figure (5) represents the results of coding image with DWT.

• **Zonal Coding Algorithm**

In this coding algorithm each image is divided into (T x V) distinct blocks, where both T and V are smaller than the image size, and as previous algorithm shows, the transform is taken for each block separately and then encoded by the Zonal algorithm.

The inverse Zonal is taken for each block and then the decoded operation is performed on each distinct block individually to finally have the output (decoded) image.

The proposed algorithm of image decoding using distinct block processing can be illustrated as follows [16]:

1. Divide (N x N) image into (T x V) blocks or (subimage) where each block is transformed and coded separately.
2. Take from the given image of (N x N) the first (T x V) blocks, so that, T and V are selected to be equal to N / \sqrt{b} , $b = 2^{2n}$, $n = 1, 2, 3, \dots, N$.
3. Transform each block (T x V) by using any of these transformation (DCT, DWT).
4. Apply Zonal coding algorithm to the transformed block as mentioned in chapter two to get the coded image block where the (variance) is used as a key code (q) to code the subimage using this formula:

$$C_T(u,v) = \begin{cases} C(u,v) - q & \text{if } C(u,v) > q \\ C(u,v) + q & \text{if } C(u,v) < q \\ 0 & \text{otherwise} \end{cases} \dots(8)$$

5. Apply inverse Zonal coding to the coded image block.
6. Take the inverse transform to the block to have the decoded image.
7. Repeat steps (3-6) until all blocks are completed.
8. The decoded image is the concatenation of all the processed distinct blocks of step (7).

The rose image was coded by using Zonal coding algorithm and the results are shown in Figure (6) that represents the results of coding the image with DCT and Figure (7) represents the results of coding the image with DWT.

5.3 A Comparison Study between Each Different Method

1. Different images were taken, each one of which was coded by threshold coding method after transforming them by DCT and the same images are taken and coded by threshold coding method after transforming them by DWT.

We compare the MSE with the two cases and we have the results shown in Table (1).

2. Different images were taken, each one of which was transformed by DCT and then coded by Zonal method and the same images are taken and transformed by DWT and then coded by Zonal method.

We compare the MSE with the two cases and we have the results shown in Table (2).

3. Different images were taken, each one of which was coded first by zonal coding method and the same images were coded by Threshold coding method, and using DWT in the two cases.

We compare the MSE with the two methods and we have the results shown in Table (3).

4. Different values of threshold in threshold method are taken, so we have following results as shown in Table (4).

5. The MSE of threshold method using both DCT and DWT that are shown in Table (4) are plotted and shown in Figure (8).

6. Different values of variance in Zonal method are taken, so we have the following results as shown in Table (5).

The MSE of Zonal Method using both DCT and DWT that are shown in Table (5) are plotted and shown in Figure (8).

6 Conclusions

Wavelet transform plays an important role in image coding.

From the work in this paper, the following points are concluded:

1. In run length coding when we take the binary image and code it. It is noted that the reconstructed binary image is exactly the same as the original binary image and there is no MSE.
2. In Huffman coding the original image is taken and coded it, its coding type is similar to the run length coding but in this type of coding the image is converted to code word and the reconstructed image is exactly the same as the original and there is no MSE.
3. In threshold coding the image is coded using two types of transforms (DCT, DWT). After coding, it is noted that using DWT gives better results than using DCT as shown in Table (1). After comparing between the MSE of the two types we have 80% of the images which have small ratio of MSE and that when DWT is compared with DCT. This makes using DWT better than using DCT.
4. In Zonal coding method we code the images using two types of transforms (DCT, DWT). After coding it is noted that using DWT gives better results than using DCT as shown in Table (2). After comparing between the MSE of the two types we have DWT better than using DCT.
5. When images are taken and transformed them by using DWT they are coded each one of which by threshold method or by Zonal method.

When comparison is made between the MSE of the two methods, it is noted that 70% of the images have a small ratio of MSE in threshold method. This means that threshold method gives better results than Zonal coding method using DWT as shown in Table (3).

6. Figure (8) represent the plot of the MSE using DCT and DWT. It is noted that the MSE using DWT lower than that of the MSE using DCT, and it's more stable.
7. Wavelet transform results need less storage spaces. When the decomposition levels increase the storage space decreases, we can use it in image compression.
8. We compare the Results of coding in different methods: RLC, DCT in Zonal method, Diagonal DWT in Zonal method, Vertical DWT in Zonal method, Horizontal DWT in Zonal method, DCT in Threshold method, Diagonal DWT in threshold method, Vertical DWT in threshold Method, and Horizontal DWT in threshold method. This is shown in Figure (9).

7. References

- [1] Robi Poliker, "The Engineering Ultimate Guide to Wavelet Analysis". [http:// www.public, Jastate.edu/nrpoliker/WAVELETS/WTtutorial/html](http://www.public.jastate.edu/nrpoliker/WAVELETS/WTtutorial/html).
- [2] Argenti, F., and Benelli, G., "IIR Implementation of Wavelet Decomposition for Digital filter Analysis" Reprinted from electronic Letters, Vol.28, PP513-515, 27th February 1992.
- [3] Shu – tao, LI and Yao – nan, WANG, "Texture Classification Using Discrete Muloi – Wavelet Transform", College of electrical and information Engineering, Hunan University, P.R. China, 2000.
- [4] Graps, A., "An Introduction to Wavelets", IEEE, Inc., Computational Science and Eng., Vol. 2, No. 2, PP 1-19, Summer 1995.
- [5] Selesnich, I.W., "Interpolation Multiwavelet Bases and Sampling Theorm", IEEE Transc. On SP, Vol. 47, PP. 1615 – 1621, 1999.
- [6] Wilson, R., "Multiresolution Image modeling", Electronic and Comm. Journal, PP. 90 – 96, April, 1997.
- [7] Goswami, J.C., and Chan, A.K., "Fundamentals of Wavelets Theory, Algorithms, and Applications", John Wily and Sons, 1999.
- [8] Hassan, K.K., "DCT / DPCM Hybrid Coding for Interlaced Image Compression", M.Sc. Thesis, University of Technology, Jan 2005.
- [9] Subhi, A., "Interframe Coding for TV Signal", M.Sc. Thesis, College of Engineering, AL-Nahrain Univ, 2000.
- [10] Umbangh, S.E., "Computer Vision and Image Processing", A practical approach using CVIP tools, Prentice Hall, Inc., Upper Saddle Saddle River, NJ 07458 (U.S.A), 1998.
- [11] Tekalp, A.M., and Pavlovic, G., "Digital Image restoration", Springer – Verlang, in A.K., Katsagglosed, New York, U.S.A., 1998.
- [12] AL-Salah, I.K.I, "Image Denoising Using Wavelet Transform", M.Sc. Thesis, College of Eng., University of Baghdad, June 2001.

- [13] Burrus, C.S., Goperath, R.A., and Guo, H., "*Introduction to Wavelet and Wavelets Transforms*", A primer Upper Saddle, NS (U.S.A), Prantice Hall, Inc., 1998.
- [14] Misite, M., Misite, Y., Oppenheim, G., and Pog., J.M., "*Wavelet Toolbox : Computation, Visualization, and Programming*", User's Guide, version 1, Mathworks, Inc., 1997.
- [15] Donoho, D.L., and Johnstone, I.M., "*Adapting to unknown smoothness via wavelet shrinkage*", Journal of American Statistic Assn., Stanford Statistics, Dep., report TR-U25, June 1995.
- [16] Abdulmunim, M.E., and Hamodi, E., "Coding Using Wavelet Transform", M.Sc. Thesis, University of Technology, Dec. 2006.

Table (1) The Comparison between DCT and DWT MSE Results using Threshold Method.

	Zonal Method	Threshold Method
	xe^{-016}	xe^{-016}
Tiger	5.6968	5.3542
Fish	5.1787	3.8271
Rose	4.2003	4.0992
Mountain	4.5926	8.1234
Sea	6.8042	6.1260
Bird	6.5655	7.8968
Kids	5.4942	5.3267
Flower	9.2481	9.1508
Nature	5.2711	7.0590
Plain	2.4116	1.8105

Table (2) The Comparison between DCT and DWT MSE Results using Zonal Method.

Threshold	MSE	
	DCT xe^{-016}	DWT xe^{-016}
0	8.1901	5.6570
- 30	8.1548	5.6573
- 60	8.1827	5.6595
- 90	8.0275	5.6658
- 120	8.0864	5.6697
- 150	8.1822	5.6709
- 180	8.1246	5.6726
- 210	9.0113	5.6719

Table (3): The Comparison between Zonal and Threshold Method MSE Results.

Threshold	MSE	
	DCT $\times 10^{-16}$	DWT $\times 10^{-16}$
0	8.1901	5.6570
- 30	8.1674	5.6575
- 60	8.1603	5.6598
- 90	8.0431	5.6658
- 120	8.1010	5.6697
- 150	8.1948	5.6709

Table (4): MSE Results for DCT and DWT using Different Threshold Values using Threshold Method

Threshold	MSE	
	DCT $\times 10^{-16}$	DWT $\times 10^{-16}$
0	8.1901	5.6570
- 30	8.1548	5.6573
- 60	8.1827	5.6595
- 90	8.0275	5.6658
- 120	8.0864	5.6697
- 150	8.1822	5.6709
- 180	8.1246	5.6726
- 210	9.0113	5.6719

Table (5): MSE Results for DCT and DWT using Different Variance Values using Zonal Method

Threshold	MSE	
	DCT $\times 10^{-16}$	DWT $\times 10^{-16}$
0	8.1901	5.6570
- 30	8.1674	5.6575
- 60	8.1603	5.6598
- 90	8.0431	5.6658
- 120	8.1010	5.6697
- 150	8.1948	5.6709

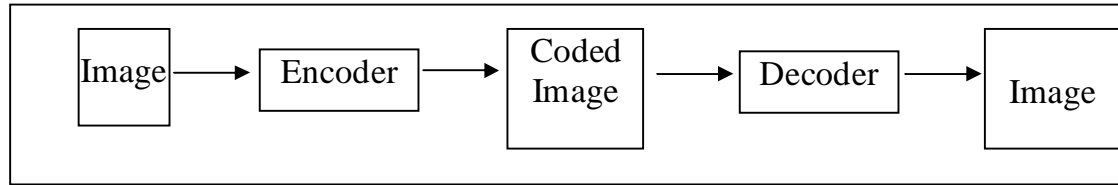


Figure (1): The Coding Process

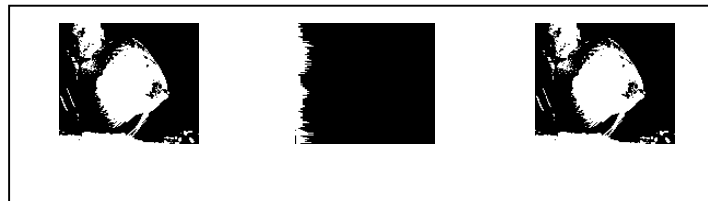


Figure (2): Results of Run Length Coding Algorithm, a) Original Binary Image. b)Coded Image c) Reconstructed Binary Image.

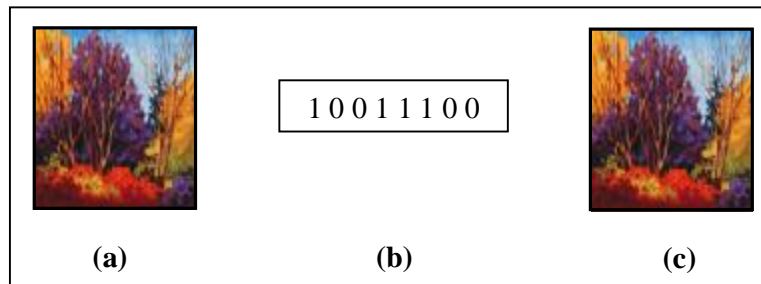


Figure (3) Results of Huffman Coding Algorithm, a) Original Image, b) Code Word, c) Reconstructed Image.

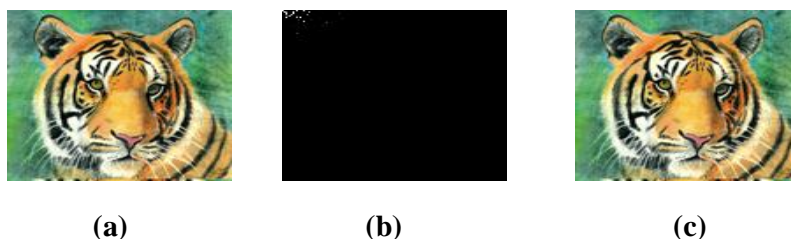
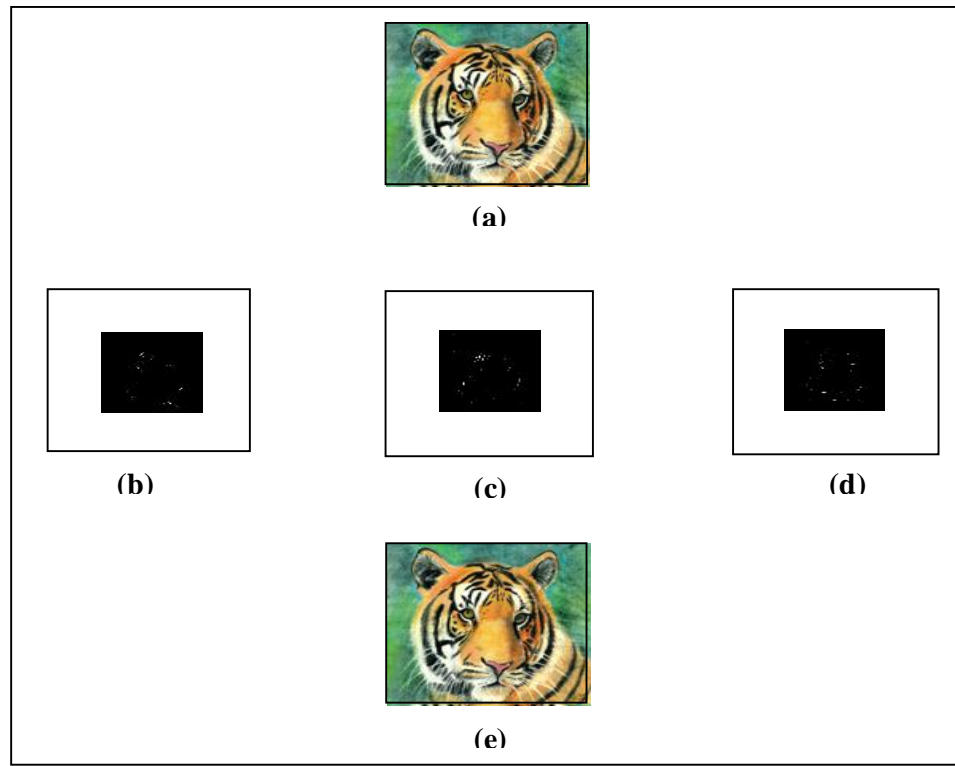


Figure (4): Results of threshold coding algorithm, a) Original image, b) Coded image using Diagonal DWT, c) Coded image using Vertical DWT d) Coded image using Horizontal DWT, e) Reconstructed image.



**Figure (5) Results of threshold coding algorithm, a) Original image,
b) Coded image using Diagonal DWT, c) Coded image using Vertical DWT d) Coded image using Horizontal DWT, e) Reconstructed image.**

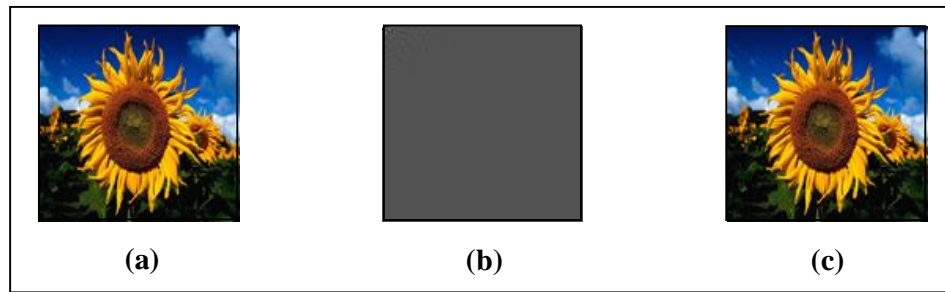


Figure (6): Results of Zonal Coding Algorithm, a) Original Image, b) Coded Image using (DCT), c) Reconstructed Image.

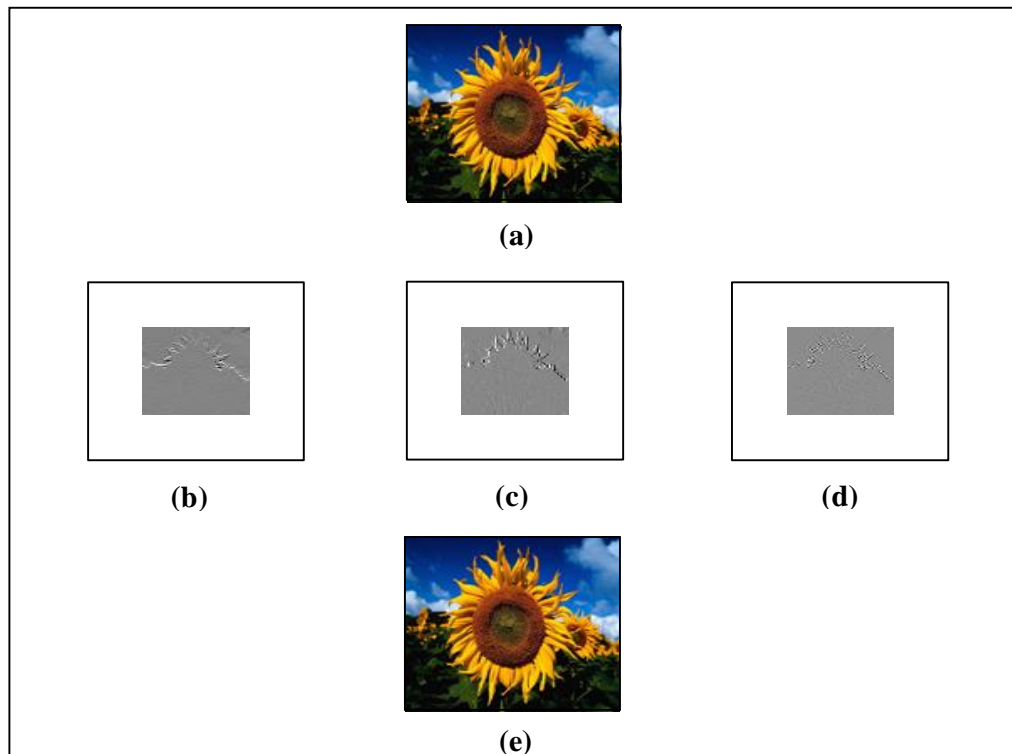


Figure (7) Results of zonal coding algorithm, a) Original image, b) Coded image using Diagonal (DWT), c) Coded image using Vertical (DWT), d) Coded image using Horizontal (DWT), e) Reconstructed image.

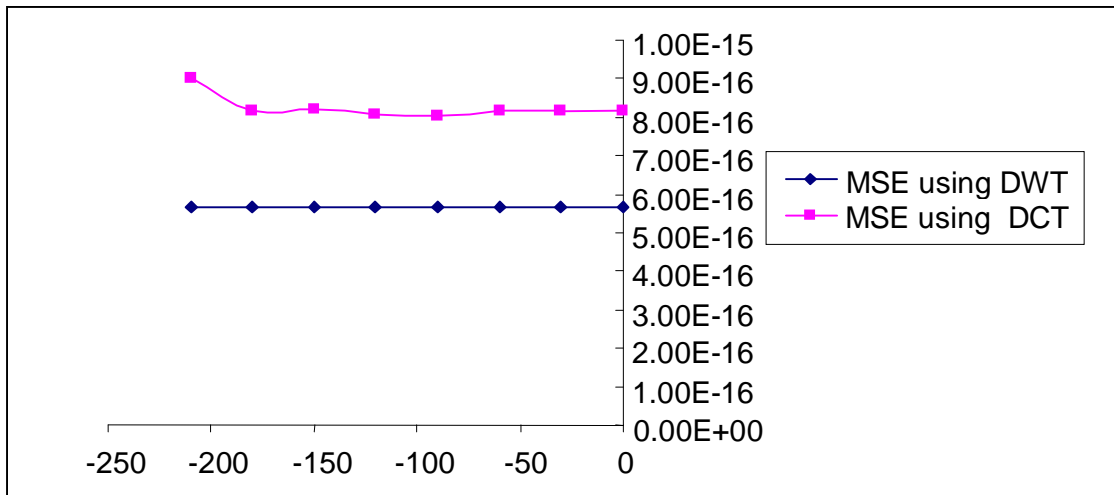


Figure (8) The Plot of MSE Results for DCT and DWT

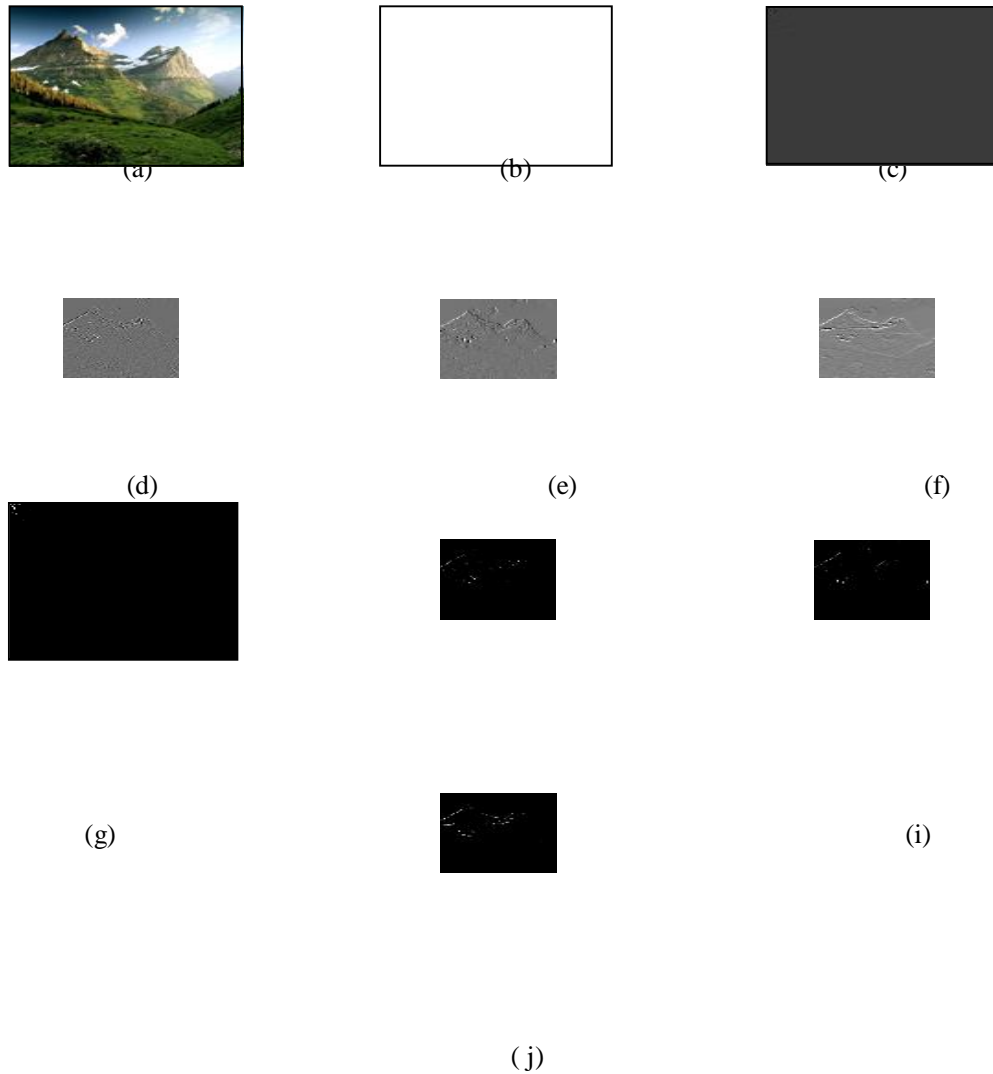


Figure (9):The Results of Coding in Different Methods: a)The Original Image, b) Coded Image using RLC, c) Coded Image using DCT in Zonal method, d) Coded Image using Diagonal DWT in Zonal method, e) Coded Image using Vertical DWT in Zonal method, f) Coded Image using Horizontal DWT in Zonal method, g)Coded Image using DCT in Threshold Method, h) Coded Image using Diagonal DWT in Threshold Method, i) Coded Image using Vertical DWT in Threshold Method, j)Coded Image using Horizontal DWT in Threshold Method.