J. of al-anbar university for pure science : Vol.2 : No.1 : 2008

ISSN: **1991-8941**

IMAGE WAVELETE COMPRESSION USING SHIFT CODING AND BITMAP SLICING

Loaye A. Goerge* **Azmi T. Huseen **** **Samyia S. Lazar****

**Baghdad university*

*** Al-Anbar University*

ABSTRACT:**The technique of image compression means reduce the size of image data in a way that can be reconstructed to its necessary components. This technique convert image data to an effective code, then this code can be decoded to reconstruct the approximate image.**

In this paper we implement image compression using wavelet transform. After loading image data, it will divide into three-color components (red, green and blue). Where each component processed independently. Each component transformed using wavelet transform to subbands (low subband and high subbands). Where the number of subbands depends on the number of transform passes. After completing the transformation process, the coding is divided into two necessary parts:

First: Process the coefficients of low subband using Differential Pulse Code Modulation (DPCM) to reduce the size of coefficients. Then these coefficients coded using S-Shift coder.

Second: Process the other subbands by dividing each subband in to eight bit slices. Then these bit sliced are coded by using chain encoder.

After completing the coding processes, the compressed data stored into file. Then these compressed data are decompressed using algorithms similar to that are used in compressed system but with inverse order.

The compression ratio in this work is up to 57 with accepted level or error and quality.

Keywords: Image wavelete , compression , shift coding , bitmap slicing

Introduction

Digital image compression based on the idea of sub-band decomposition or discrete wavelet transform. The Discrete Wavelet Transform (DWT) is one of the most important lossy image compression methods [1]. Since many of the wavelet transform coefficients for a typical image tend to be small or zero, these coefficients can be easily coded. Thus wavelet transforms are a useful tool for image compression [2].

Image compression based on wavelet transform has recently received an increasing interest [3]. This scheme first involves the transformation of data into more suitable form. The goal of transformations is a compact, complete representation of the image. The transform should decorrelate the specially distributed energy into fewer data samples such that no information is lost. Compression occurs in the second step when the transformed

images, usually those with insignificant energy levels are discarded. This step, called quantization process, is not invertable. The coding process typically followed the quantization stage. Entropy coding minimizes the redundancy in the bit stream and is fully invertable at the decoding end [4]. The DWT lends itself to image compression for the following reasons

A. Many natural image exhibit fractal or self-similar behavior. The DWT is an efficient representation for these signals since all wavelet coefficients can be computed from the wavelet coefficients at one scale.
B. DWT is well suited to progressive encoding due to its pyramid decomposition.
C. The space-scale decomposition in the DWT allows for the incorporation of HVS characteristic [4].

Many wavelet-based compression schemes have been reported in the literature, ranging from simple entropy coding to complex coding techniques such as tree structures coding, and some modern wavelet-based compression methods such as CREW and SPIHT. All these schemes can, actually, be described in general form that the compression includes forward wavelet transform, quantizer, and lossless entropy coders. The corresponding decompressor is formed by lossless entropy decoder, de-quantizer, and inverse wavelet transform. Wavelet based compression has good compression results in both rate and distortion sense [1].

Wavelet Packet Transform (WPT)

The Wavelet Packet Decomposition (WPD) method is a generalization of wavelet decomposition that offers a richer range of possibilities for signal analysis [5]. Unlike the DWT, which only decomposes the low frequency components, the WPD utilizes both the low frequency components and the high frequency components. For j-level decomposition, in the case of DWT there are j+1 possible ways to decompose or encode the signal, while in WPD there are 2j different ways to encode the signal. A WPD is actually a full binary tree structure, which separate the frequency into sections of same length. However, this scheme results in additional computational complexity. Wavelet packet functions are designed by generalizing the filter bank tree that relates wavelet and conjugate mirror filters [6, 7]. For each set of N coefficients, we obtain two coefficient sets of N/2 length after processing by WPT. The number of coefficient sets is 2m if the original coefficient set is processed for m resolution.

In summary the wavelet packet has a number of useful properties:

A. It can represent smooth function.
B. It has unconditional basis function; the choice of wavelet basis should usually do according to reasonable consideration(s).

The potential of wavelet packet lies in its capacity to offer a rich menu of basis functions that satisfy perfect reconstruction conditions, from which the best one can be chosen for a given application.

In This paper we use two types of Packet Decomposition:

1- Packet Decomposition with Haar Wavelet Transform .

2- Packet Decomposition with Integer Wavelet Transform (IW- Tap 3/5) .

Suggested Image Compression System:

The aim of this system includes developing and applying an efficient image compression system. The proposed compression system focuses on an innovative scheme for adaptive wavelet coding technique. Where the Low Low (LL) subband of the transformed image is processed by Differential Pulse Code Modulation (DPCM), and then coded by S-Shift coder. The other subbands are uniformly quantized and then coded by using bitmaps coding. The proposed compression system implies some control coding parameters; the effects of these parameters were investigated to determine the suitable range for each one of them. Figure (1) shows a block diagram of the suggested compression and decompression schemes..

This suggested scheme could be divided into two general modules:

First// Encoder Module

This module involves the following steps:

1. Loading Image Data (Reading BMP file) into three RGB layers:

2. Applying the Forward Wavelet Transform:

The wavelet transform can decompose the approximation and the detail subbands of the decomposed image (packet decomposition) to produce:

Total number of subbands = 4^ number of level

The following wavelet transform are used:

A. Packet Decomposition with Haar Wavelet Transform:

Algorithm list (1) shows the steps of decomposition of an image using Haar transform (packet scheme). Figure (2) shows the transformed Girl image with 2 and 3 levels of packet decomposition using Haar transform .See the details in the references [8,9,10].

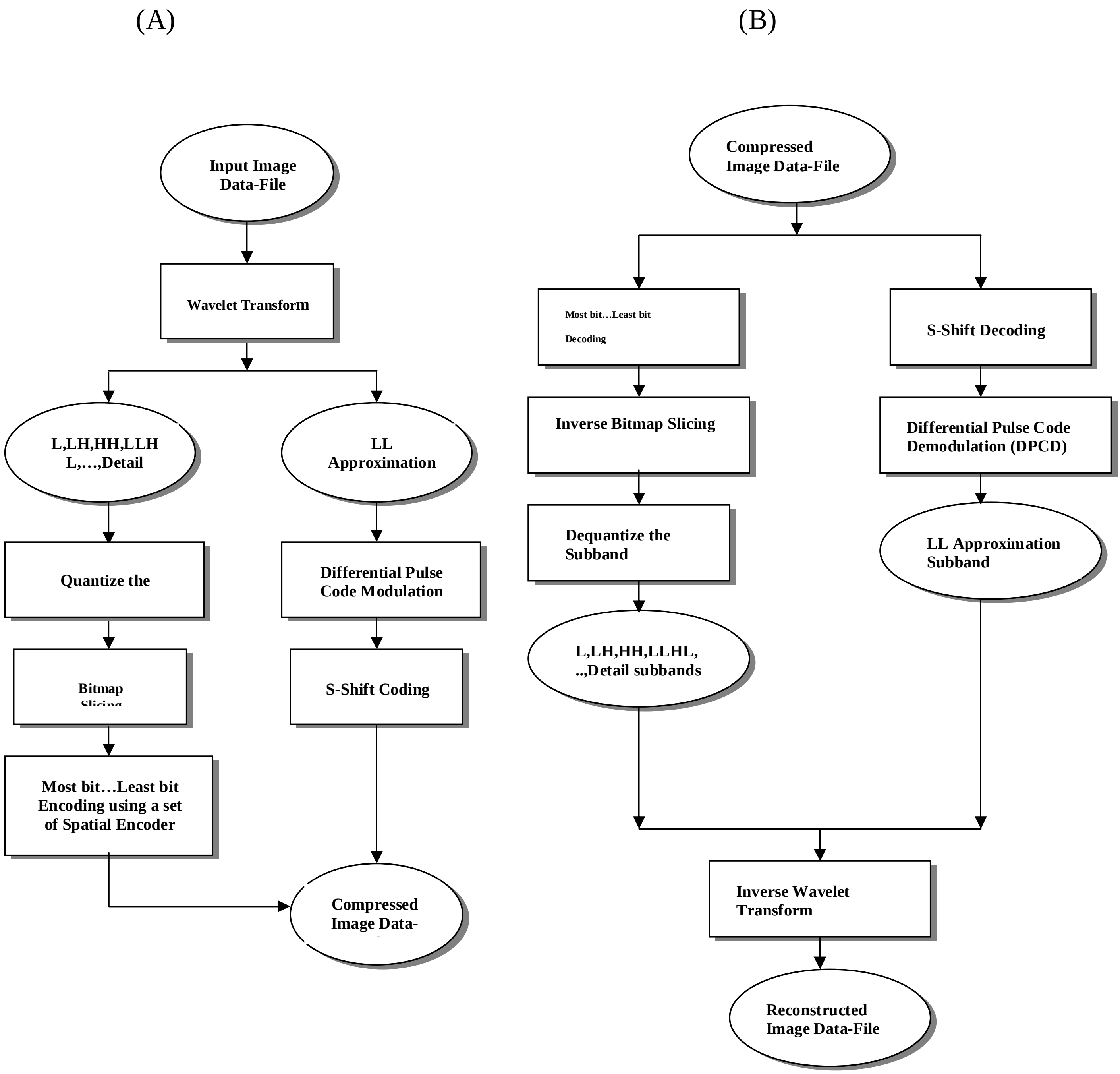


Figure (1).The suggested image compression system (A) and reconstruction system (B) based on based on wavelet transform

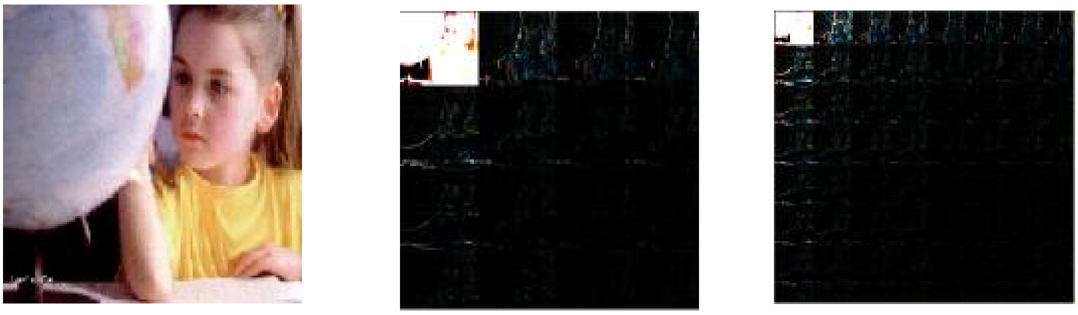


Figure (2). The transformed "girl" image using packet decomposition scheme based on Haar transform

B. Packet Decomposition with Integer Wavelet Transform (IW- Tap 3/5)

In the suggested system, the integer wavelet transform of type (Tap3/5) was also used. This transform was applied on the three color components of the image, and the calculation was separated into steps according to the width and height of the image. Figure (3) Show pepper image using 2 and 3 levels of decomposition using packet (IW Tap 3\5). .See the details in the references [8-10]. Figure (4) illustrate the implementation steps of image decomposition using IWT-Tap3/5.

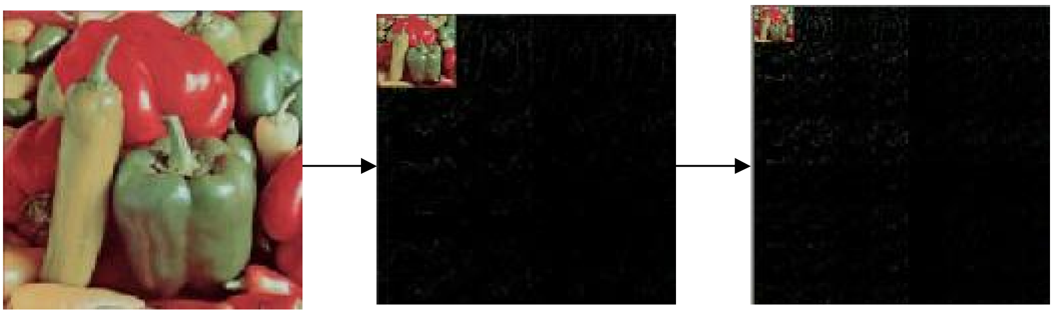


Figure (3) The transform of "Pepper " image using packet decomposition based on tap3/5 wavelet transform

Algorithm (1) Packet image decomposition using Haar wavelet transform

(1) Apply the following operation to specify the beginning of data and width of image at each subband.

1.1) Ofs[0]=0; Siz[0]=W; Lay[0]=1; Ns=0; J=0; /* Ofs[] is start of data, Siz[] is size of data, Lay[] is level of transform, Ns is number of subband.

1.2) While J £ Ns do:

If Lay[J] < Nopass Then: W=(Siz[j]+1)\2; I=Lay[J]+1;

Ns=Ns+1; Ofs[Ns]=Ofs[J]; Siz[Ns]=W; Lay[Ns]=I;

Ns=Ns+1; Ofs[Ns]=Ofs[J]+W; Siz[Ns]=Siz[J]-W;Lay[Ns]=I

End If; J=J+1;

1.3) End While;

/* Apply Haar filters (packet) on Rows of C[] */

(2) Apply the following operation for each subband

2.1) Set Xs=Ofs[I]; W=Siz[I]; Xe=Xs+W-1;

2.2) Perform the following operations if W is even:

2.2.1) Set Wh=W\2; Whm=Wh-1; X1=Xs;

2.2.2) For X=0 To Whm: X2=X1+1; XL=Xs+X;XH=XL+Wh

2.2.3) For Y=0 To H-1 do:

D[XL,Y]=C[X1,Y]+C[X2,Y];D[XH,Y]=C[X1,Y]-C[X2,Y]

2.2.4) End For Y loop;

2.2.5) Set X1=X1+2;

2.2.6) End For X loop;

Continue

2.3) Perform the following operations if W is odd:

2.3.1) Set Wh=(W+1)\2;Whm=Wh-1;Whm2=Whm-1;X1=Xs

2.3.2) For X=0 **To** Whm2: X2=X1+1; XL=Xs+X; XH=XL+Wh

2.3.3) For Y=0 **To** H-1 do:

D[XL,Y]=C[X1,Y]+C[X2,Y]; D[XH,Y]=C[X1,Y]-C[X2,Y]

2.3.4) End For Y **loop**;

2.3.5) Set X1=X1+2;

2.3.6) End For X **loop**;

2.3.7) Set XL=Xs+Whm;

2.3.8) For Y=0 **To** H-1do: D[XL,Y]=2*C[Xe,Y];

(3) Apply the following operation to specify the beginning of data and height of image subband.

3.1) Ofs[0]=0; Siz[0]=Hgt; Lay[0]=1; Ns=0; J=0;

3.2) While J ≤ Ns do:

If Lay[J] < Nopass Then: H= (Siz[j]+1)\2; I=Lay[J]+1;

Ns=Ns+1; Ofs[Ns]=Ofs[J]; Siz[Ns]=H; Lay[Ns]=I;

Ns=Ns+1; Ofs[Ns]=Ofs[J]+H;Siz[Ns]=Siz[J]-H; Lay[Ns]=I

End If; J=J+1;

3.3) End While;

/ Apply Haar filters (packet) on Column of C[] */*

(4) Apply the following operation for each subband

4.1) Set Ys=Ofs[I]; H=Siz[I]; Ye=Ys+H-1;

4.2) Perform the following operations if H is even:

4.2.1) Set Hh=H\2; Hhm=Hh-1; Y1=Ys;

4.2.2) For Y=0 **To** Hhm: Y2=Y1+1;YL=Ys+Y; YH=YL+Hh

4.2.3) For X=0 **To** W-1 do:

D[X,YL]=C[X,Y1]+C[X,Y2];D[X,YH]=C[X,Y1]-C[X,Y2]

4.2.4) End For X **loop**;

4.2.5) Set Y1=Y1+2;

4.2.6) End For Y **loop**;

4.3) Perform the following operations if H is odd:

4.3.1) Set Hh=(H+1)\2;Hhm=Hh-1; Hhm2=Hhm-1;Y1=Ys;

4.3.2) For Y=0 **To** Hhm2:Y2=Y1+1; YL=Ys+Y; YH=YL+Hh

4.3.3) For X=0 **To** W-1 do:

D[X,YL]=C[X,Y1]+C[X,Y2];D[X,YH]=C[X,Y1]-C[X,Y2]

4.3.4) End For X **loop**;

4.3.5) Set Y1=Y1+2;

4.3.6) End For Y **loop**;

4.3.7) Set YL=Ys+Hhm;

4.3.8) For X=0 **To** W-1: D[X,YL]=2*C[X,Y];

3. Encoding the pproximation subband coefficients:

After the wavelet transform, the compression process is divided into two major parts. The first part applied on the approximation subband, while the second part is applied on the detailed subbands. Encoding of approximation is performed by applying Differential Pulse Code Modulation (DPCM) to reduce the range of the coefficients values. Since the range is small, they will need small number of bits for representation when it is compared with the original values. Algorithm list (2) shows the steps of DPCM encoder. The output of the DPCM is fed to the S-Shift coding method to produce the compressed bit streams for LL subband.

Algorithm (2) Differential Pulse Code Modulation (DPCM) encoder

(1) Inputs: Wsb; Hsb; C[] /* where Wsb and Hsb are the width and height of the LL subband and C[] is array of LL coefficients*/
(2) Initialization: DPCM [0]= C [0, 0]; B= C[0,0]; I=0
(3) For Y=0 **To** Hsb do:
(4) Perform the following operations if Y is even:
  4.1) For X = 0 **To** Wsb do:
  4.2) Increment I by 1;
  4.3) Subtract the value of current coefficient from the previous one:
    DPCM [I] = C [X, Y]–B; B = C [X, Y];
  4.4) End For (X) **loop**;

(5) Perform the following operations if Y is odd:
  5.1) For X = Wsb **Downto** 0 do: **Increment** I by 1;
  5.2) Subtract the value of current coefficient from the previous one:
    DPCM [I] = C [X, Y] – B; B = C [X, Y];
  5.3) End For (X) **loop**;
(6) End For Y **loop**

3.1) S-Shift Coding Method This method is used to reduce the size (in bits) of the output of the DPCM coding stage. The idea of this method is to encode the sequence of numbers by code words whose bit length is less than the bit length required to represent the maximum coefficient value, that may produced by DPCM encoder, when it is coded by using fixed length coding. The size (in bits) required to encode this sequence using fixed length coding can be calculated by using the following equation: Size = r * n --------------------(1) Where: (r) is the number of bits required to

encode the maximum number existing in the sequence, (n) is the number of elements. Figure (5) illustrate the way of selection the s-shift encoder.

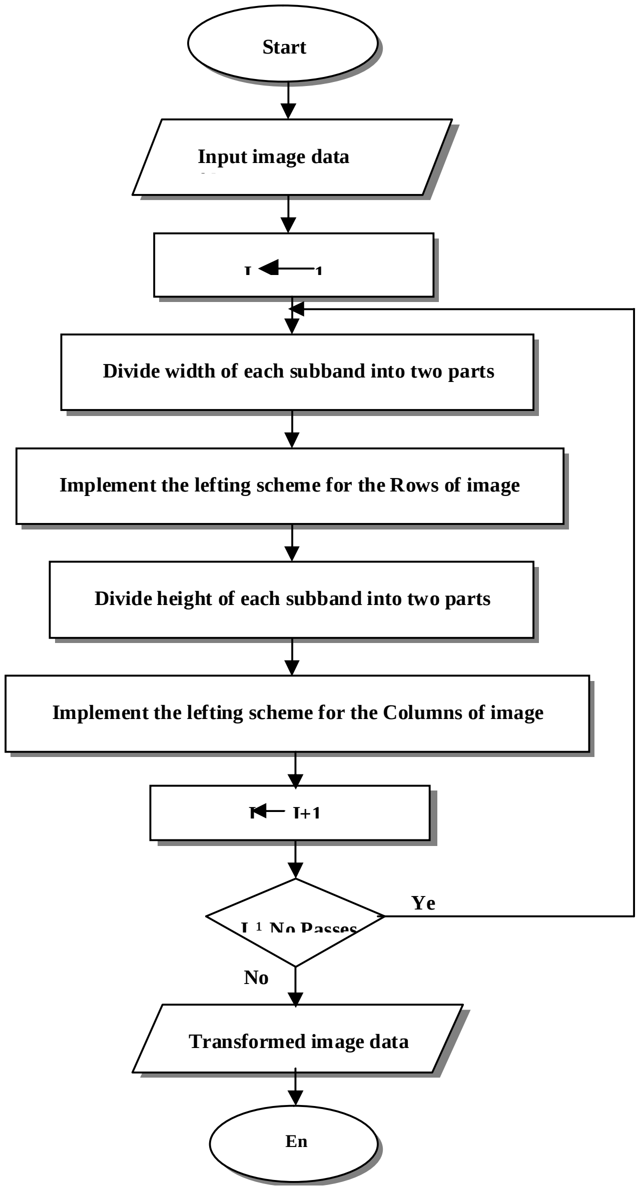


Figure (4) packet decomposition using lefting scheme with (IW-Tap 3/5)

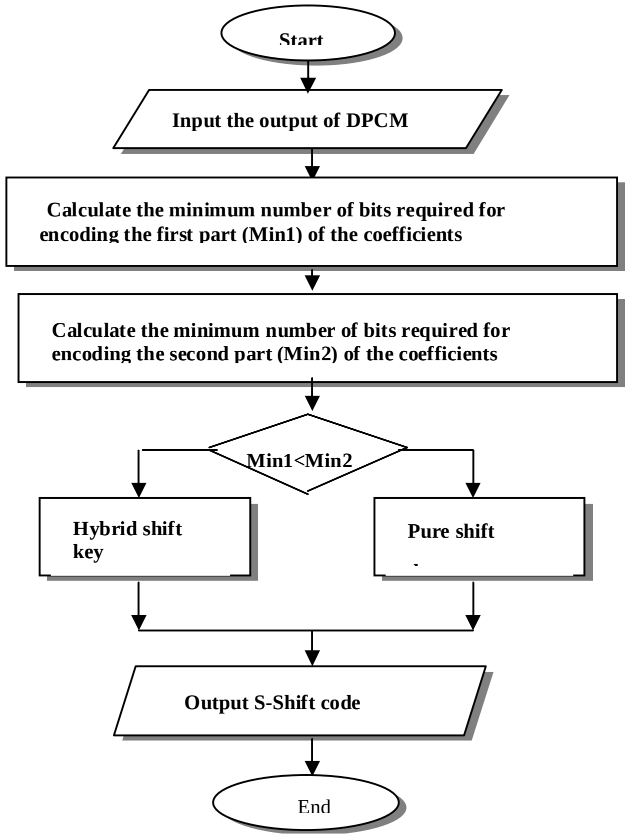


Figure (5) Flowchart for select S-Shift encoder

Encoding the detail subbands coefficients:

At first, the detail coefficients are uniformly quantized using certain quantization step. Second, the output of this process is sliced into bit-planes by using the bit slicing process.

Bit Slicing

The quantized wavelet coefficients are coded by using bit-plane. The coding is progressively done from the most significant bit-plane to least significant bit-plane.

The basic idea behind using this method is based on decomposing the quantized transform coefficients into binary components (i.e., bitmap layers), such that each layer represents a (0-1) bitmap, for example each point in ith bitmap layer represents the ith bit value of the corresponding transform coefficient. The resulting layers may different in their spatial and statistical characteristics. If these characteristics are considered to choose the suitable coding method and to encode each layer separately, the resulting code size may, probably, be compact (i.e., has a small size) in compression with that of the original image, consequently the image compression task will be achieved.

This section illustrates the adopted representation (modeling) of the transform coefficient bits. It describes how the coefficients are arranged into bit-planes. First the sign of quantized detail coefficients are expressed in a sign bit-plane by splitting the sign of each coefficient representing it as bit state. The sign bit is taken (1) for the positive (or zero) coefficients and (0) for the negative coefficients. The magnitude for each particular subband, there is a maximum number of magnitude bits. Thus, it is important to know the maximum value of the transformed coefficients for each subband in order to determine the maximum number of bits needed to represent each

Algorithm (3.) Bitmap slicing encoder

(1) Input size of subband (Sband), LowBit, subband coefficients C[]
(2) Split the sign from coefficient magnitude and put in Sign[]
  2.1) For each (X, Y) coefficients in subband of size Sband do:
    If C[X, Y] $\geq$ 0 **then** Sign[X, Y]=0: **Else** Sign[X, Y]=1;
  2.2) End For each X & Y;
(3) Find maximum(Max) value in subband to determine maximum number of bits (HiBit) needed to represent each subband.
  3.1) Initialize Max to the first coefficients in C[]
  3.2) For each (X, Y) coefficients in subband of size Sband
    If Max < C[X, Y] **Then** Max=C[X, Y]
    If Max > 0 **Then** B= ln(Max) / ln(2); HiBit=Tranc[B];
  3.3) For each (X, Y) coefficients in subband of size Sband do:
  3.4) For I=LowBit **To** HiBit do:
    Bt = $2^{\wedge I}$; J = C[X, Y] AND Bt;
  3.5) Save J in BitPlan[I]
  3.6) End For I **loop**;
  3.7) End For each X & Y

subband. Algorithm list (3) illustrates the implemented steps of bit-slicing coding.

Chains Coder

Chain coders are used in the suggested system to represent the higher bit-planes, which contain connected sequences of straight-line streams of specific length and direction. Typically, this representation is based on 4-connectivity or 8-connectivity of segments. In the proposed scheme, 8-connectivity was used to represent the contents of some higher bit-planes. The direction of each segment is coded by using a numbering scheme such as the one shown in figure (6). Digital images usually are acquired and processed in grid format with equally spacing in the x-and y-direction. So, a chain code could be generated by following the connected points in, say, clockwise direction and assigning a direction to the segment connecting every pair of connecting ones. This method generally is unacceptable for two principal reasons: (1) the resulting chain of codes tends to be quite long, and (2) any small disturbances along the connected chains due to noise or imperfect segmentation cause changes in the code. Algorithm (4) illustrates the implemented steps of the chain coding [8].

Second\\Decoding Module

The decoding process is performed to reconstruct the image data. The decoder uses the compressed bit-stream (produced by the encoder module). Decoding process converts the compressed image data to a reconstructed image data. Thus, we can define the decoder as an embodiment of the decoding process. In the proposed scheme, all steps of decoder are performed on each of the three-color components (i.e., red, green, and blue), then these three components are merged to reconstruct an output image. The decoding process consists of two parts, and apply the backward wavelet transform.

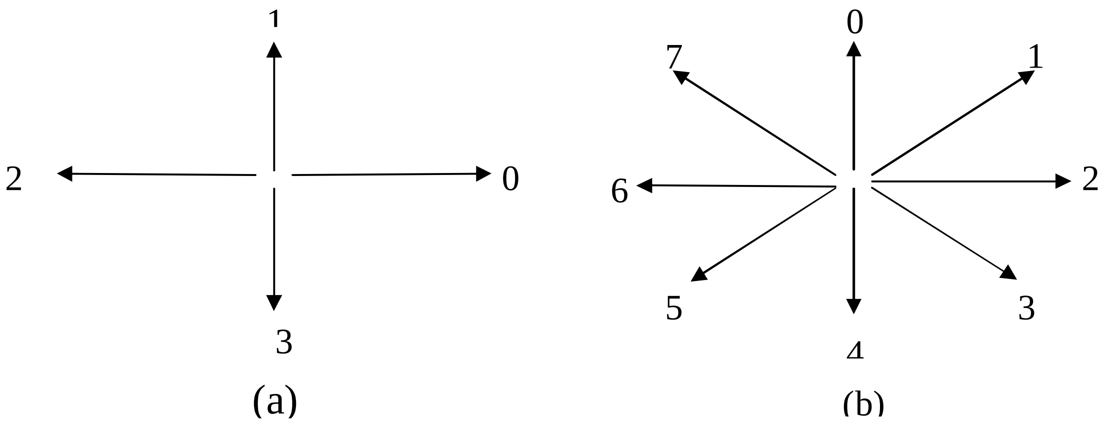


Figure (6) Chain coding directions for (a) 4-direction code, (b) 8-direction code

Algorithm (4): chain coding

(1) Input BitSlic[],WHsb /BitSlic is bit plane, WHsb is subband size/
(2) For each (X,Y) in the block of size WHsb do:
2.1) If BitSlic(X,Y) = 1 **then**:
2.1.1) Store the location (X,Y): StoreX = X; StoreY = Y;
2.1.2) Flag = 1; N = 0 /* N is the length of chain*/
2.1.3) Repeat the following operations:
2.1.4) Set X1 = X+1; X2 = X-1; Y1 = Y+1; Y2=Y-1;
If BitSlic(X,Y2)=1 **Then**:
Flag=0;**Output=0**;N=N+1;Y=Y2
Else if BitSlic(X1,Y2) = 1 **Then**
Flag=0; **Output=1**; N=N+1; X=X1; Y=Y2;
Else if BitSlic(X1,Y) = 2 **Then**
Flag=0; **Output=2**; N=N+1; X=X1;
Else if BitSlic(X1,Y1) = 1 **Then**
Flag=0; **Output=3**; N=N+1; X=X1; Y=Y1;
Else if BitSlic(X,Y1) = 1 **Then**
Flag=0; **Output=4**; N=N+1; Y=Y1;
Else if BitSlic(X2,Y1) = 1 **Then**
Flag=0; **Output=5**; N=N+1; X=X2; Y=Y1;
Else if BitSlic(X2,Y) = 1 **Then**
Flag=0; **Output=6**; N=N+1; X=X2;
Else if BitSlic(X2,Y2) = 1 **Then**
Flag=0; **Output=7**; N=N+1; X=X2; Y=Y2;
End if
Until Flag =1;
2.2) End If
(3) End For each X & Y;

1-Decoding of the approximation subband:

This process begins by applying the S-Shift decoding method. Then, its output is fed to the Differential Pulse Code Demodulation (DPCD) to reconstruct the LL subband of the transformed image.

composed to reconstruct the sequence of numbers which represent the coefficients indices, these indices are given their sign value from the stored sign bit plane. The process of inverse bit slicing can be described by the following equation:

$$X_i = \sum_{n=0}^{M} b_i(n) 2^n \quad ,.......................................(2)$$

Where: X_i represents the ith coefficient index, $b_i(n)$ represents ith bit in the bit plane (n), and n represents the number of the considered bit plane.

Where the initialization and input are same as that listed in algorithm (3). At last each coefficient is multiplied by the quantization step size (note, these coefficients are not exactly equal to the original coefficients because of quantization).

<u>Experimental Results</u>

In this section some test results and the corresponding reconstructed images produced by applying the suggested image compression system are presented. Advantages and drawbacks of the system are discussed on the basis of the presented results.

Tests on two natural images (pepper and girl) were performed by using the suggested compression system. The dimension of the considered images is 256×256. These images are of type bitmap files (.BMP) and their pixel format is (24 bit/pixel).

2. Decoding of the detail subband:

The start step for decoding the details is the chain decoding to reconstruct the bit planes (the reconstructed bit planes are exactly similar to the original bit planes). In the next stage the constructed bit planes will

The maximum partition size was taken to power of two. This will give the largest possible number of level of decomposition; the largest number of levels is the number of times the original block dimensions can be divided by two and produce integer result.

<u>Test Results</u>

The effects of a number of parameters have been investigated to evaluate the performance of the suggested wavelet based image compression scheme. The fidelity measures used in our test results are the Mean Square Error (MSE), Mean Absolute Error (MAE), Signal to Noise Ratio (SNR), Pick Signal to Noise Ratio (PSNR), and the Normalized Mean Square Error (NMSE). Also the compression ratio is calculated.

1 -The Effect of the Wavelet Transform Type

In this section the effects of the type of wavelet transform are introduced. The test results indicated that this parameter has little effect on the compression performance and, consequently, on the reconstructed image. Table (1) and figure (7) present the effect of transform type on the quality of the reconstructed pepper and girl images, and on the compression ratio.

2- The Effect of the Quantization Step-Size

The test results indicated that the quantization step-size has significant

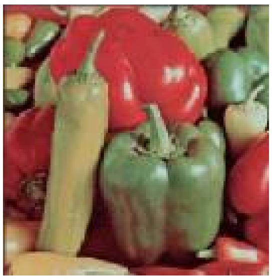
influence on the compression performance and, consequently, on quality of the reconstructed image. The increase in quantization step-size will lead to degradation in the reconstructed image but will cause an increase in the compression ratio. In the proposed coding scheme different quantization steps values were used for each sub-band. The change in values of the quantization step depends on the history of partitioning applied to produce the considered sub-band. Table (2) and figure (8) present the effect of the initial quantization step size on the quality of the reconstructed pepper and girl images, and on the compression ratio. The results show that the quantization step size has a great effect on the compression ratios and it is causes a monotonic increase in compression ratio as the magnitude of the initial quantization step size is increased.

3 - The Effect of Number Wavelet Levels

The levels of wavelet decomposition of an image have a great effect on the reconstructed image quality and also on the compression ratio. In the previous discussion, all examples imply the use of same number of decomposition levels, but the magnitude of the levels are increased to show its effect on the compression performance. Table (3) and Figure (9) present the effect of the number of decomposition levels on the Compression factor.

Table (1) The effect of the wavelet transform type on the performance of the suggested image compression system

Test Images	**Type of WT**	**MSE**	**MAE**	**SNR**	**PSNR**	NMSE	**CR**	**No.bit LL**	**No.bit HH**
Pepper	***Packet Haar***	2.684	1.154	41.914	45.987	4.578E-05	4.434	122903	293849
	Packet tap3\5	9.122	3.392	36.601	43.330	9.155E-05	4.561	108105	246745
Girl	***Packet Haar***	3.545	1.409	43.412	45.382	6.104E-05	3.403	119711	342576
	Packet tap3\5	10.86	3.894	38.552	42.952	3.052E-05	3.601	117089	319863



Packet Harr wavelet
No.transform pass=1
CR=3.405

Packet IW Tap 3/ 5wavelet
No.transform pass=1
CR=3.601

Packet Harr wavelet
No.transform pass=1
CR=4.434

Packet IW Tap 3/ 5wavelet
No.transform pass=1
CR=4.561

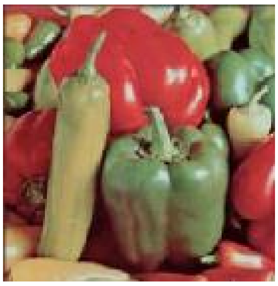
Figure (7) The reconstructed images from table (1)

Table (2) The effect of the initial quantization step size (Qunt) on the compression performance factors

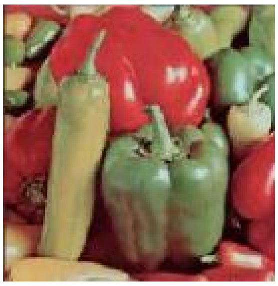
Test Images	**Qunt**	**MSE**	**MAE**	**SNR**	**PSNR**	NMSE	**CR**	**No.bitLL**	**No.bit HH**
pepper	1	2.863	1.960	41.633	45.846	0	3.026	108105	411877
	2	9.122	3.392	36.601	43.330	9.155E-05	4.434	108105	246745
	3	19.387	5.339	33.327	41.693	1.984E-04	5.639	108105	170878
	4	34.035	6.970	30.882	40.471	8.698E-04	7.997	108105	88629
	5	39.578	7.513	30.227	40.144	1.358E-03	9.057	108105	65269
	6	48.537	8.233	29.341	39.700	2.029E-03	10.15	108105	46892
	7	57.650	8.842	28.594	39.327	2.640E-03	11.12	108105	33427
	8	64.540	9.244	28.103	39.081	4.318E-03	11.90	108105	24100
girl	1	8.063	2.248	39.845	43.698	3.815E-04	2.520	117089	507030
	2	10.86	3.894	38.552	42.952	3.052E-05	4.601	117089	319863
	3	21.626	5.711	35.560	41.456	1.831E-04	4.658	117089	220650
	4	36.820	7.415	33.249	40.300	5.951E-04	6.298	117089	132705
	5	45.081	8.183	32.370	39.861	1.358E-03	7.352	117089	96901
	6	57.228	9.136	31.334	39.343	2.228E-03	8.696	117089	63835
	7	66.761	9.778	30.665	39.008	3.006E-03	9.632	117089	46251
	8	75.377	10.28	30.138	38.745	4.196E-03	10.52	117089	32487

Table (3) The effect of number of decomposition levels on the compression performance factors

Test Image	No.WT levels	MSE	MAE	SNR	PSNR	NMSE	CR	No.bit LL	No.bit HH
Pepper	1	9.122	3.392	36.601	43.330	9.16E-05	4.434	108105	246745
	2	16.990	5.131	33.900	41.980	6ز12E-04	10.59	28377	120129
	3	24.916	6.678	32.237	41.148	7.02E-04	21.74	8045	64342
	4	32.498	7.713	31.083	40.571	1.52E-03	36.73	2112	40717
	5	39.677	8.564	30.216	40.138	2.15E-03	47.72	534	32434
	6	246.33	1.299	22.286	36.173	0.189682	52.68	143	29721
	7	253.74	1.677	22.158	36.109	0.191009	54.56	45	28791
	8	268.11	14.82	21.918	35.989	0.196808	55.02	18	28576
Girl	1	10.86	3.894	38.552	42.952	3.052E-05	3.601	117089	319863
	2	19.045	5.641	36.112	41.731	1.678E-04	10.03	27182	129645
	3	26.247	6.789	34.719	41.035	3.51E-04	22.33	7973	62489
	4	33.381	7.670	33.675	40.513	7.94E-04	35.42	2147	42272
	5	40.958	8.540	32.787	40.069	1.07E-03	43.98	562	35214
	6	49.933	9.440	31.926	39.639	1.49E-03	47.28	143	33134
	7	56.987	10.11	31.352	39.352	1.53E-03	48.54	40	32374
	8	72.424	11.15	30.311	38.831	9.92E-03	49.12	18	32010



Qunt=1
No.transform pass=1
CR=3. 026

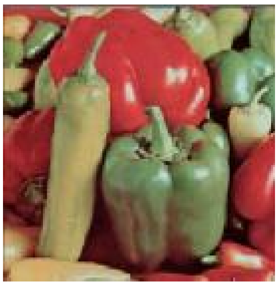


Qunt=2
No.transform pass=1
CR=4. 434

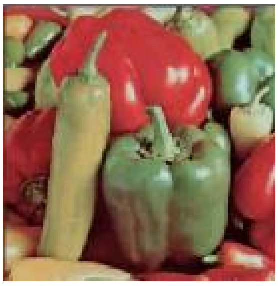
Qunt=3
No.transform pass=1
CR=5. 639

Qunt=4
No.transform pass=1
CR=7. 997

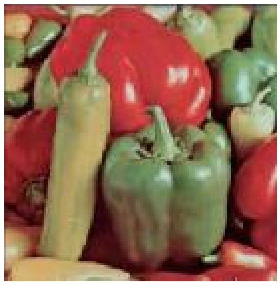
Qunt=5
No.transform pass=1
CR=9. 075



Qunt=6
No.transform pass=1
CR=10. 15



Qunt=1
No.transform pass=1
CR=11. 12



Qunt=8
No.transform pass=1
CR=11. 90

Qunt=1
No.transform pass=1
CR=2. 520

Qunt=2
No.transform pass=1
CR=4. 601

Qunt=3
No.transform pass=1
CR=4. 658

Qunt=4
No.transform pass=1
CR=6. 298

Qunt=5
No.transform pass=1
CR=7. 352

Qunt=6
No.transform pass=1
CR=8. 696

Qunt=7
No.transform pass=1
CR=9. 632

Qunt=8
No.transform pass=1
CR=10. 52

Figure (8) the reconstructed images from table (2)…(cont.)

Qunt=2
No.Transform Pass=1
CR=4.434

Qunt=2
No.Transform Pass=2
CR=10.59

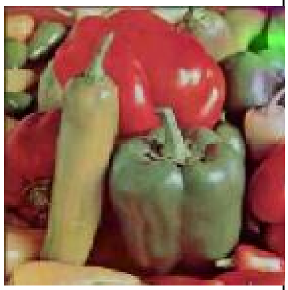
Qunt=2
No.Transform Pass=3
CR=21.74

Qunt=2
No.Transform Pass=4
CR=36.73

Qunt=2
No.Transform Pass=5
CR=47.72

Qunt=2
No.Transform Pass=6
CR=52.56

Qunt=2
No.Transform Pass=7
CR=54.02



Qunt=2
No.Transform Pass=8
CR=55.02

Qunt=2
No.Transform Pass=1
CR=3.601

Qunt=2
No.Transform Pass=2
CR=10.03

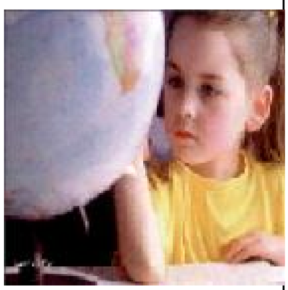
Qunt=2
No.Transform Pass=3
CR=22.33

Qunt=2
No.Transform Pass=4
CR=35.42

Qunt=2
No.Transform Pass=5
CR=43.98

Qunt=2
No.Transform Pass=6
CR=47.28

Qunt=2
No.Transform Pass=7
CR=48.54



Qunt=2
No.Transform Pass=
CR=49.12

Figure(9):The reconstructed images from table(3) …Cont

Conclusions:

The tests are performed to study the effects of some control compression parameters on the system performance. These derived conclusions are the following:

1-Wavelet based image compression is ideal for adaptive compression since it is inherently a multi-resolution scheme. Variable levels of compression can be easily achieved.

3- The use of bi-orthogonal wavelet transform (Tap3/5) with other encoding methods can lead to compression ratio between (1- 57) with acceptable range of error and PSNR.

4-Wavelet packet transform leads to positive effects on the suggested compression performance.

5-The selection of quantization step size should be done carefully, because test results indicated that this factor has significant effect on the compression ratio and PSNR factor.

6-The used chain encoder have an effect on the proposed scheme by reducing its complexity and speed-up the compression time, while the use of vector quantization and DPCM lead to an increase in the compressed image quality (i.e., increasing PSNR).

References

[1] Al-Kaphaje G. K., "Adaptive Fractal Image Compression", MSc. thesis, National Computer Center, Higher Education Institute of Computer and Informatic, Sep. 2001.

[2] Adams M.D., "Reversible Integer-to-Integer Wavelet Transform for Image Coding", Ph.D. thesis, Department of Electrical and Computer Engineering, The University of British, Columbia, Sep. 2002.

[3] Chair Ch.Chen, "On the Selection of Image Compression Algorithms", Department of Computer Science, National Tsing Hua University, August 1998.

[4] Lyer L.R., "Image Compression Using Balanced Multi-wavelet", MSc. Thesis, Electrical Engineering, Virginia Polytechnic Institute and State University, June 2001. http://www.amara.com/current/wavelet.html

[5] Misiti M., Misiti Y., Oppenheim G., and Poggi J.M., "Wavelet Toolbox: For Use with MATLAB", By the MathWorks, Inc, 1997.

[6] Vian S. K. Al-Doori, "Adaptive Discrete Filter for Based on Wavelet Packet Transform", MSc. thesis, Collage of Engineering, Saddam University, 1998.

[7] Aqiel N. Z. Al_ma'amoory, "Perfomance Evaluation of adaptive Filters based on the Wavelet Transform", Collage of Engineering, Saddam University, October 2001.

[8] Gonzalez R.C., and Woods R.E, "Digital Image Processing", by Prentice Hall, Inc., 2002.

[9] Sue Twelves, Mike Wu, and Andrew White, "JPEG2000 Wavelet Transform Using StarCore", by Motorola, Inc, 2001.

[10] Musa Ali K., "Watermark Application in Color Images Using Wavele Transforms", Ph.D. thesis, Informatics Institute for Postgraduate Stydies, Feb. 2004

ضغط الصور باستخدام التحويل المويجي

لؤي ادور جورج عزمي توفيق حسين الراوي سامية شمعون لازار
E.mail: scicolanb@yahoo.com

الخلاصة

إن تقنية ضغط الصور تعني تقليل حجم بيانات الصور بحيث يمكن استعادة اهم مكوناتها. هذه التقنية تعمل على تحويل بيانات الصور الى ترميز مشفر وهذا الترميز بالامكان فكه فيما بعد ليعطي صورة مقربة للصورة الأصلية.في هذه الدراسة طبقت تقنية ضغط الصور باستخدام طريقة التحويل المويجي (Wavelet Transform) على الصور الملونة (RGB) بمستوياتها اللونية حيث يتم تحويل كل مستوى وبأستخدام طريقة التحويل الموجي الى مستويات متعددة من الحزم الفرعية Subbands (عالية و واطئة)، حيث إن عدد الحزم الفرعية Subbands يعتمد على عدد مستويات التحويل الموجي.

بعد إكمال عملية التحويل فأن العمل سوف يجزء الى جزئين أساسيين :

الأول : يعتمد على معالجة معاملات الحزم الفرعية الواطئة بحيث تطبق طريقة (Differential Pulse Code Modulation) لتقليل مستوى القيم لهذه المعاملات ومن ثم تشفير البيانات بأستخدام طريقة الترميز بالتزحيف(S-Shift).

الثاني : يعمل على معالجة الحزم الفرعية العالية حيث يعمل على تقسيم كل حزمة الى مستويات ذات الثمانية بت باستخدام عملية تقطيع البت (Bit Slicing) ثم شفرت النتائج باستخدام طريقة Chain encoder .

البيانات المضغوطة الناتجة من عملية الضغط يتم فتح ضغطها باستخدام خوارزميات مناظرة لتلك المستخدمة في نظام الضغط ولكن بتسلسل معاكس. يمكن تمثيل البيانات المضغوطة في هذا البحث بنظام ضغط غير متناظر .كانت نسبة الانضغاط المتحققة في العمل يصل الى 57 مع ظهور نسبة خطأ مقبولة ونوعية مقبولة ايضاً .