# A Fuzzy Neural Networks based EZW Image Compression System

Dr. S. Abdul Khader Jilani
Lecturer
College of Computers & IT
University of Tabuk, Tabuk, The
Kingdom of Saudi Arabia

Dr. S. Abdul Sattar
Professor & Dean of Academics
Royal Institute of Tech & Science
JNT University,Kukatpally, Hyderabad,
India

## ABSTRACT

The transport of images across communication paths is an expensive process. The limitation in allocated bandwidth leads to slower communication. To exchange the rate of transmission in the limited bandwidth the Image data must be compressed before transmission. JPEG2000 image compression system follows huffman coding for image compression. Embedded zero tree wavelet (EZW) coding exploits the multi-resolution properties of the wavelet transform when compared to existing wavelet transforms. Artificial Neural Network has been applied to many problems in image processing and has demonstrated their superiority over classical methods when dealing with noisy or incomplete data for image compression applications. A fuzzy optimization design based on neural networks is presented as a new method of image processing. The combination system adopts a new fuzzy neuron network (FNN) which can appropriately adjust input and output values, and increase robustness, stability and working speed of the network by achieving high compression ratio.

## General Terms

Artificial Neural Networks, Fuzzy Logic, Image Compression.

## Keywords

Bandwidth, Compression, EZW, JPEG2000, Transmission.

## 1. INTRODUCTION

Digital imagery has an enormous impact on industrial, scientific and computer applications. It is no surprise that image coding has been a subject of great commercial interest in today's world. Uncompressed digital images require considerable storage capacity and transmission bandwidth. Efficient image compression solutions are becoming more critical with the recent growth of data intensive, multimedia-based web applications.

Various research works were carried out on both lossy and lossless image compression. The JPEG committee released a new image-coding standard, JPEG2000 that serves the enhancement to the existing JPEG system. The JPEG2000 implements a new way of compressing images based on the wavelet transform in contrast to the Discrete Cosine Transformation (DCT) used in JPEG standard.

The problem of image compression is more important in many applications, particularly for progressive transmission, image browsing, multimedia applications, and compatible transcoding [1] for multiple bit rates. A majority of today's internet bandwidth is estimated to be used for images and video transmission. Recent multimedia applications for handheld and portable devices place a limit on the available wireless bandwidth. The bandwidth is limited even with new connection standards. Wavelet based image compression techniques such as JPEG 2000 offers more compression than conventional methods in terms of compression ratio.

Very few works on generation of an accurate bit stream that claims higher PSNR performance at rates between 0.25 and 1 bit/pixel were made. Z. Xiong, K. Ramchandran and M. Orchad [2] uses a tree coding where the said value is zero if its energy is less than perceptually based threshold. An approach to the low bit rate image transform coding is presented by S.Mallat and F.Falzon [1]. The overview to lossy wavelet image compression for JPEG 2000 [3] and wavelet transformation with embedded zerotree coding were presented by Bryan E. Usevitch [4]. Colm Mulcahy presented the application of embedded coding on an isolated tile image for image compression in his paper [5] using haar wavelet. The paper gives an approach towards embedded coding using haar wavelet transform for real images.

'The embedded zerotree wavelet coding for image processing' presented by Shapiro in his paper [6] uses embedded coding for image compression. The embedded zerotree wavelet coding [4], [6] uses the wavelet coefficient [7], [8] for encoding. Wavelets use the multiresolution analysis [8] for decomposing the image into sub-band [9] as detail and approximate coefficients. In subband coding systems [9], the coefficients from a given image are extracted. The JPEG2000 coding system [3] & [4] presents image compression using wavelet transform. These papers present the JPEG 2000 coding system architecture for still image compression.

For improving the ANN efficiency for image compression various works were suggested in past. In [10] the author expounds the principle of BP neural network with applications to image compression and the neural network models. Then an image compressing algorithm based on improved BP network is developed. Wavelet-based image compression [11] provides substantial improvements in picture quality at higher compression ratios. The author in [12] suggests that the classic image compression techniques such as JPEG and MPEG have serious limitations at high compression rate; the decompressed image gets really fuzzy or indistinguishable.To overcome this problem, artificial neural network techniques are used where he propose a bipolar sigmoidal backpropagation BBP algorithm to train a feed forward auto associative neural network. The idea of author in [13] is that a neural network could be trained to recognize an optimum ratio for DCT compression of an image upon presenting the image to the network. The neural network associates the image intensity with its compression ratios in search for an optimum ratio. Feedforward networks using backpropagation algorithm [14] adopting the method of steepest descent for error minimization is popular and widely adopted and is directly applied to image compression. Image data compression using Vector Quantization (VQ) has received a lot of attention because of its simplicity and adaptability. VQ requires the input image to be processed as vectors or blocks of image pixels. In [15] the author presents an image compression scheme that uses the wavelet transform and neural network. Firstly, image is decomposed at different scales by using the wavelet transform. Then, the different quantization and coding schemes for each sub-image are carried out in accordance with its statistical properties and distributed properties of the wavelet coefficients.

The author [16] shows that on the basis of pre-serving monotonicity an improved algorithm of image compression and reconstruction can be proposed. The algorithm is based on partitioning the range of the function fI which corresponds to an image I.A new image compression algorithm that combines a fuzzy technique and wavelet transform [17] is presented. This algorithm cuts down the computational costs in deriving the wavelet coefficients using a Haar wavelet transform to decompose the image, and a Fuzzy Logic technique to arrange the wavelet coefficients. Furthermore, it utilizes the Huffman or LZW algorithm to compress the fuzzified coefficients. A novel fuzzy learning vector quantization algorithm [18] for image compression was developed and consists of two basic issues. Firstly, a modified objective function of the fuzzy c-means algorithm is reformulated and then is minimized by means of an iterative gradient-descent procedure. Secondly, the training procedure is equipped with a systematic strategy to accomplish a smooth transition from fuzzy mode, where each training vector is assigned to more than one codebook vectors, to crisp mode, where each training vector is assigned to only one codebook vector.

A method of fuzzy optimization design based on neural networks [19] is presented as a new method of image processing. The combination system adopts a new fuzzy neuron network (FNN) which can appropriately adjust input and output, and increase robustness, stability and working speed of the network. Besides FNN, wavelet transform also be applied to compression algorithm for a higher and controllable compression rate.

## 2. EMBEDDED ZERO TREE WAVELET CODING

Embedded Zerotree Wavelet encoder is based on progressive encoding to compress an image into a bit stream with increasing accuracy. When more bits are added to the stream, the decoded image is contain more detail of the image a property similar to JPEG encoded images.

Coding an image using the EZW scheme, together with some optimizations results in a remarkably effective image compression with the property that the compressed data stream can have any bit rate desired. Any bit rate is only possible if there is information loss somewhere so that the compression is lossy. However, lossless compression is also possible with an EZW encoder, with less optimal results. Figure 1 below shows the block diagram for the EZW coding system.

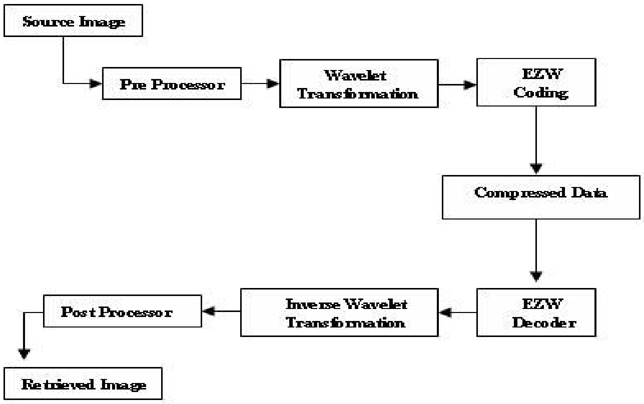


**Figure 1 . Block diagram for the EZW coding system**

### 2.1 The Algorithm

The EZW output stream starts with information to synchronize the decoder. The minimum information required by the decoder for its functionality is the number of wavelet transforms levels used and the initial threshold, basically a constant levels (3) of wavelet transform were used for transformation. The first step in the EZW coding algorithm is to determine the initial threshold. The initial threshold $t_0$ is given as $t_0=2^{[\log_2(MAX(|\gamma(x,y)|))]}$.Where MAX($|\gamma(x,y)|$) means the maximum coefficient value in the image and $\gamma(x,y)$ denotes the coefficient. Then taking the obtained threshold as the initial value the scaled sub-band samples are been passed for dominant pass and subordinate pass. Under each pass the threshold is decreased by half the value. This comparison is carried out until the threshold reaches to the minimum threshold, the algorithm implemented as presented below.

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

**Figure 2 . EZW algorithm**

The EZW encoder is based on two important observations: 1) Natural images in general have a low pass spectrum. When an image is wavelet transformed the energy in the subbands decreases as the scale decreases (low scale means high resolution), so the wavelet coefficients is, on average, be smaller in the higher subbands than in the lower subbands. This shows that progressive encoding is a very natural choice for compressing wavelet-transformed images, since the higher subbands only add detail parameter. 2) Large wavelet coefficients are more important than small wavelet coefficients.

These two observations are used for encoding the wavelet coefficients in decreasing order, in several passes. For every pass a threshold is chosen against which all the wavelet coefficients are measured. If a wavelet coefficient is larger than the threshold it is encoded and removed from the image, if it is smaller it is left for the next pass. When all the wavelet coefficients have been visited the threshold is lowered and the image is scanned again to add more detail to the already encoded image. This process is repeated until all the wavelet coefficients have been encoded completely.

## 2.2 Dominant Pass

The image is scanned and a symbol is returned for every coefficient. If the coefficient is larger than the threshold a **P** (positive) is coded. If the coefficient is smaller than negative of threshold an **N** (negative) is coded. If the coefficient is the root of a zerotree then a **T** (zerotree) is coded and finally, if the coefficient is smaller than the threshold but it is not the root of a zerotree, then a **Z** (isolated zero) is coded. This happens when the coefficient larger than the threshold in the subtree. All the coefficients that are in positive value, larger than the current threshold are extracted and placed without their sign on the subordinate list and their positions in the image are filled with zeros. This prevents them from being coded again. The dominant pass can thus be implemented as;

```
initialize_fifo();
while (fifo_not_empty) {
    get_coded_coefficient_from_fifo();
    if coefficient was coded as P, N or Z then {
        code_next_scan_coefficient();
        put_coded_coefficient_in_fifo();
        if coefficient was coded as P or N then {
            add abs(coefficient) to subordinate list;
            set coefficient position to zero;
        }
    }
}
```

**Figure 3 . Dominant Pass**

## 2.3 Subordinate Pass

The dominant pass is always followed by a subordinate pass where the coded data get coded in 1 or 0 to be transmitted, the process for subordinate pass is as illustrated;

```
subordinate_threshold = current_threshold/2;
for all elements on subordinate list do {
    if coefficient>subordinate_threshold then {
        output a one;
        coefficient = coefficient-subordinate_threshold;
    }
    else output a zero;
}
```

**Figure 4 . Subordinate Pass**

A wavelet transform transforms a signal from the time domain to the joint time-scale domain. i.e. the wavelet coefficients are two-dimensional. To compress the transformed signal not only the coefficient values, but also their position in time has to be coded. When the signal is an image then the position in time is better expressed as the position in space. After wavelet transforming an image it can be represented using trees because of the subsampling that is performed in the transform. A coefficient in a lower subband can be thought of as having four descendants in the next higher subband as shown in the below FIGURE 5. The four descendants each also have four descendants in the next higher subband, which gives a quad-tree, with every root having four leafs. A zerotree is defined as a quad-tree of which all nodes are equal to or smaller than the root and the root is smaller than the threshold against which the wavelet coefficients are currently being measured. The tree is coded with a single symbol and reconstructed by the decoder as a quad-tree filled with zeroes. The EZW encoder codes the zerotree based on the observation that wavelet coefficients decrease with scale. In a zerotree all the coefficients in a quad tree are smaller than the threshold if the root is smaller than this threshold. Under this case the whole tree can be coded with a single zerotree (T) symbol.

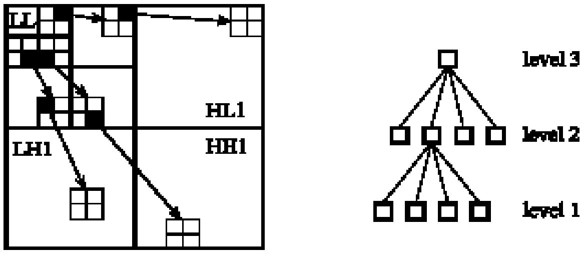
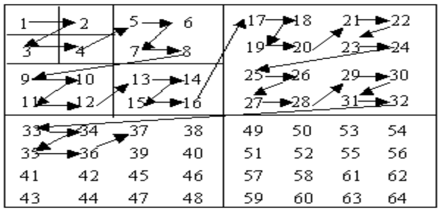


**Figure 5 . The inter relationship of multiple levels in wavelet scaled image**

EZW encoding uses a predefined scan order to encode the position of the wavelet coefficients. Through the use of zerotree many positions are encoded implicitly. Several scan orders are possible, as long as the lower subbands are completely scanned before going on to the higher subbands. The relations between wavelet coefficients in different subbands, and there scan path is show in Figure 5 and Figure 6.



| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 5 | 6 | 17 | 18 | 21 | 22 |
| 3 | 4 | 7 | 8 | 19 | 20 | 23 | 24 |
| 9 | 10 | 13 | 14 | 25 | 26 | 29 | 30 |
| 11 | 12 | 15 | 16 | 27 | 28 | 31 | 32 |
| 33 | 34 | 37 | 38 | 49 | 50 | 53 | 54 |
| 35 | 36 | 39 | 40 | 51 | 52 | 55 | 56 |
| 41 | 42 | 45 | 46 | 57 | 58 | 61 | 62 |
| 43 | 44 | 47 | 48 | 59 | 60 | 63 | 64 |

**Figure 6 . Scan order for embedded coding**

## 3. FUZZY BACKPROPAGATION TRAINING

Let the configuration of fuzzy Backpropagation be l-m-n.

Step 1: Generate the initial weight sets $\tilde{W}$ for the input-hidden layer where each $\tilde{W}_{ji} = \left(W_{mji}, W_{\alpha ji}, W_{\beta ji}\right)$ is an LR-type fuzzy number. Also generate the weight set $\tilde{W}'$ for the hidden-output layer where $\tilde{W}'_{kj} = \left(W'_{mkj}, W'_{\alpha kj}, W'_{\beta kj}\right)$.

Step 2: Let $\left(\tilde{I}_p, D_p\right), p = 1,2,...,N$ be the N input-output pattern set that fuzzy BP needs to be trained with. Here, $\tilde{I}_p = \left(\tilde{I}_{p0}, \tilde{I}_{p1}, ..., \tilde{I}_{pl}\right)$ where each $\tilde{I}_{pi}$ is an LR-type fuzzy number, i.e. $\tilde{I}_{pi} = \left(\tilde{I}_{pmi}, \tilde{I}_{p\alpha i}, \tilde{I}_{p\beta i}\right)$. $D_p$ is a crisp output.

Step 3: Let ITRNS denote the number of iterations. Set the counters for the number of iterations and number of pattern sets to be trained to zero.

i.e. COUNT_OF_ITRNS=0; p=1;

Step 4: Assign values for $\eta$ and $\alpha$

Initialize:

$$\tilde{\Delta} W(t-1) = 0;$$

$$\tilde{\Delta} W'(t-1) = 0;$$

Also, $\tilde{W}(t-1) = 0$ and $\tilde{W}'(t-1) = 0$

Step 5:Get next pattern set $\left(\tilde{I}_p, D_p\right)$.

Assign $\tilde{O}_{pi} = \tilde{I}_{pi}, i = 1,2,...,l; \tilde{O} = (1,0,0)$ for the input neurons.

Step 6: Compute

$$O'_{pj} = f\left(NET_{pj}\right), j = 1,2,...,m; O'_{po} = 1$$

where $NET_{pj} = CE\left(\sum_{i=0}^{l} \tilde{W}_{ji}\tilde{O}_{pi}\right)$ for the hidden neurons.

Step 7: Compute

$$O''_{pk} = f\left(NET'_{pj}\right), k = 0,1,2,...,n-1$$

where $NET'_{pk} = CE\left(\sum_{j=0}^{m} \tilde{W}_{kj} O'_{pj}\right)$ for the output neurons.

Step 8: Compute change of weights $\tilde{\Delta} W'(t)$ for the hidden-output layer as follows:

8.1 Compute

$$\nabla E_p(t) = \left(\frac{\partial E_p}{\partial W'_{mkj}}, \frac{\partial E_p}{\partial W'_{\alpha kj}}, \frac{\partial E_p}{\partial W'_{\beta kj}}\right)$$

Where

$$\frac{\partial E_p}{\partial W'_{mkj}} = -\left(D_{pk} - O''_{pk}\right)O''_{pk}\left(1 - O''_{pk}\right).1.O'_{pj}$$

$$\frac{\partial E_p}{\partial W'_{\alpha kj}} = -\left(D_{pk} - O''_{pk}\right)O''_{pk}\left(1 - O''_{pk}\right).\left(\frac{-1}{3}\right).O'_{pj}$$

$$\frac{\partial E_p}{\partial W'_{\beta kj}} = -\left(D_{pk} - O''_{pk}\right)O''_{pk}\left(1 - O''_{pk}\right).\left(\frac{1}{3}\right).O'_{pj}$$

8.2 Compute

$$\tilde{\Delta} W'(t) = -\eta\nabla Ep(t) + \alpha\tilde{\Delta} W'(t-1).$$

Step 9: Compute change of weights $\Delta\tilde{W}(t)$ for the input-hidden layer as follows:

9.1

Let $\delta_{pmk} = -\left(D_{pk} - O''_{pk}\right)O''_{pk}\left(1 - O''_{pk}\right).1$

$$\delta_{p\alpha k} = -\left(D_{pk} - O''_{pk}\right)O''_{pk}\left(1 - O''_{pk}\right).\left(\frac{-1}{3}\right)$$

$$\delta_{p\beta k} = -\left(D_{pk} - O''_{pk}\right)O''_{pk}\left(1 - O''_{pk}\right).\left(\frac{1}{3}\right)$$

Compute

$$\nabla E_p(t) = \left(\frac{\partial E_p}{\partial W_{mji}}, \frac{\partial E_p}{\partial W_{\alpha ji}}, \frac{\partial E_p}{\partial W_{\beta ji}}\right) \text{ where}$$

$$\frac{\partial E_p}{\partial W_{mji}} = \left(\sum_k \delta_{pmk} W'_{mkj}\right)O'_{pj}\left(1 - O'_{pj}\right).1.\tilde{O}_{pi}$$

$$\frac{\partial E_p}{\partial W_{\alpha ji}} = \left(\sum_k \delta_{p\alpha k} W'_{\alpha kj}\right)O'_{pj}\left(1 - O'_{pj}\right).\left(\frac{-1}{3}\right).\tilde{O}_{pi}$$

$$\frac{\partial E_p}{\partial W_{\beta ji}} = \left(\sum_k \delta_{p\beta k} W'_{\beta kj}\right)O'_{pj}\left(1 - O'_{pj}\right).\left(\frac{1}{3}\right).\tilde{O}_{pi}$$

9.2 Compute

$$\Delta\tilde{W}'(t) = -\eta\nabla Ep(t) + \alpha\tilde{\Delta} W'(t).$$

Step 10: Update weights for the input-hidden and hidden-output layers as

$$\tilde{W}(t) = \tilde{W}(t-1) + \Delta\tilde{W}(t)$$

$$\tilde{W}'(t) = \tilde{W}'(t-1) + \Delta\tilde{W}'(t)$$

Step 11: p=p+1;

If $(p \leq N)$ goto Step 5;

Step 12: COUNT_OF_ITRNS=COUNT_OF_ITRNS+1:

If COUNT_OF_ITRNS < ITRNS

{ Reset pointer to first pattern in the training set;

p=1;

goto Step 5;

}

Step 13: Output $\tilde{W}$ and $\tilde{W}'$ the final weight sets.

## 4. FUZZY BACKPROPAGATION INFERENCE

Let $\tilde{F}_p, p = 1,2,...,N'$ be the patterns whose output values are to be inferred.

Let $\tilde{W}$ and $\tilde{W}'$ be the weight sets obtained after training fuzzy backpropagation.

Step 1: p=1;

Step 2: Get next pattern $\tilde{F}_p$ ;

Step 3: Compute

$$\tilde{O}_{pi} = \tilde{F}_{pi}, i = 1,2,...,l$$

$$\tilde{O}_o = (1,0,0);$$

for the input neurons.

Step 4: Compute

$$O'_{pj} = f(NET_{pj})j = 1,2,...,m;$$

$$O'_{po} = 1$$

Where $NET_{pj} = CE\left(\sum_{i=0}^{l} \tilde{W}_{ji}\,\tilde{O}_{pi}\right)$

for the hidden neurons.

Step 5: Compute

$$O''_{pk} = f(NET'_{pj}), k = 0,1,2,...,n-1$$

Where $NET'_{pk} = CE\left(\sum_{j=0}^{m} \tilde{W}_{kj}\,O'_{pj}\right)$

for the output neurons.

Step 6: Output the associated output

$$O''_{pk}, k = 0,1,2,...,n-1$$

Step 7: p=p+1;

If $(p \leq N')$ goto Step 2;

## 5. APPROACH

The design unit implements the Embedded Zerotree Wavelet coding system with Fuzzy Backpropagation Networks for data compression. The coding system reads the multiresolution component of the image obtain from the transformation module and pass the data to the decoder unit to retrieve the image back. Figure 7 below shows the implemented embedded zero tree wavelet coding system with fuzzy optimization design based on neural networks for image processing.

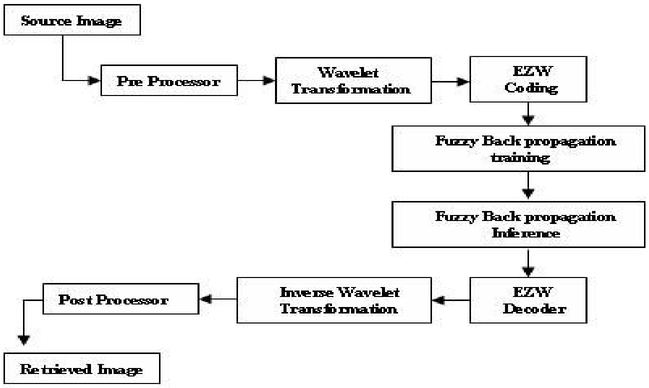


**Figure 7 . Block diagram for the proposed EZW coding system with Fuzzy Backpropagation Networks**

Before the processing of image data the image are preprocessed to improve the rate of operation for the coding system. Under preprocessing tiling of the original image is carried out. The term “tiling” refers to the partition of the original (source) image into rectangular nonoverlapping blocks (tiles), which are compressed independently, as though they were entirely distinct images. All operations, including component mixing, wavelet transform, quantization and entropy coding are performed independently on the image tiles. Tiling reduces memory requirements, and since they are also reconstructed independently, they can be used for decoding specific parts of the image instead of the whole image. All tiles have exactly the same dimensions, except may be those at the boundary of the image. Arbitrary tile sizes are allowed, up to and including the entire image (i.e., the whole image is regarded as one tile). This unit transforms the input image from time domain to frequency domain and decomposes the original image into its fundamental components.

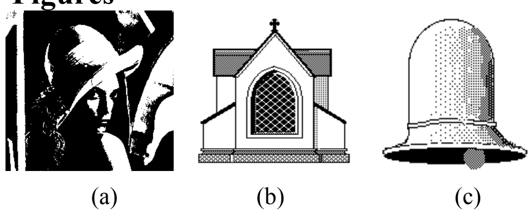
The wavelet transform uses filter banks for the decomposition of preprocessed original image into 3 details and 1 approximate coefficient. The filtering is carried out by convolving the input image with the filter coefficients passed. The Embedded Zerotree Wavelet(EZW) encoder encodes the decomposed image by recognizing the priority of decomposed image pixel. The encoder module calculates a initial threshold for coding given by $T_0$ = 2^($\log_2 x_{max}$). The encoding process is performed using 2 passes namely dominant pass and subordinate pass. The dominant pass scan the coefficient using the given threshold. These passes are repeated for n cycles reducing the current threshold by 2 until the required data bit rate is reached.

After the image is broken down into a set of values, it is necessary to alter them into a form ready for use with the FNN. A file is prepared where these set of values are written into two identical vectors to form a fuzzy backpropagation training pair. In other words, the first vector would be the input vector and the second vector would be the desired output. As many of the images tested were extremely large, there were many similar training pairs. The fuzzy backpropagation inference is then applied as shown in section 4 for image compressionto eliminate the redundant information which also reduces the transmission time of FNN. Finally, the output produced by the FNN in the form of decompressed vectors or windows, is reconstructed and then given as input to EZW decoder.

The decoding unit reconstructs the image by identifying these set of values. Inverse transformation is the process of retrieving back the image data from the obtained image values. The image data transformed and decomposed under encoding side is rearranged from higher level decomposition to lower level with the highest decomposed level been arranged at the top. Finally, the reconstruction of the obtained decomposed component into their proper graphical representation, and are then compared with the original image.

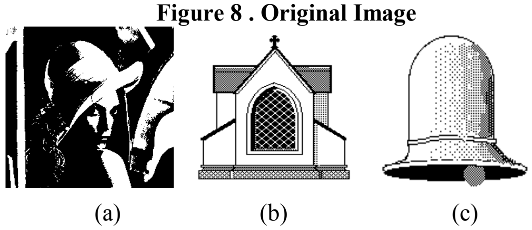
# 6. RESULTS

## 6.1 Figures



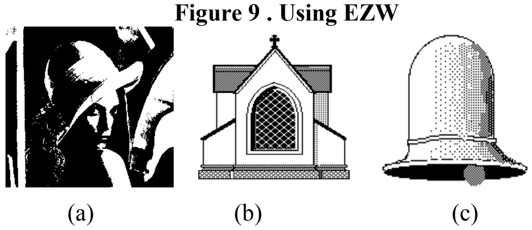
(a) (b) (c)

**Figure 8 . Original Image**



(a) (b) (c)

**Figure 9 . Using EZW**



(a) (b) (c)

**Figure 10 . Using EZW with FNN**

## 6.2 Tables

**Table 1 . Error Rate Comparison Table**

| **Error Rate Comparison** | **Sample 1 (256 x 256** | **Sample 2 (184 x 152** | **Sample 3 (128 x 144** |
|---|---|---|---|
|  | **Lena.tif)** | **Church.tif)** | **Bell.tif)** |
| EZW | 2.6 | 1.83 | 1.02 |
| EZW with FNN | 2.34 | 1.35 | 0.61 |

# 7. CONCLUSION

It is observed that embedded zerotree wavelet coding with fuzzy backpropagation networks is able to achieve good performance with relatively less computational effort. EZW with fuzzy backpropagation learning algorithm does not require complicated bit allocation procedures like subband coding does, and it does not require prior knowledge of the image source like JPEG does (to optimize quantization tables). EZW also has the desirable property, resulting from its successive approximation quantization.

The desirable result of an embedded bit stream is that it is very easy to generate coded outputs with the exact desired size for generating the training pairs. Discarding of the coded output stream does not produce visual artifacts. Since the truncation only eliminates the least significant refinement bits of coefficients rather than eliminating entire coefficients as is done in subband coding.

Embedded zerotree wavelet coding with fuzzy backpropagationartificial networks shows approximately 30% more accuracy in retrieving image compare to the existing EZW coding system.

# 8. REFERENCES

[1] S.Mallat and F.Falzon, "Analysis of low bit rate image transform coding," IEEE Trans. Signal Processing, vol. 46, pp. 1027-1042,Apr. 1998.

[2] Z. Xiong, K. Ramachandran, and M. Orchad, "Space-frequency quantization for wavelet image coding," IEEE Trans. Signal Processing, vol. 6, pp. 677 - 693, May. 1997.

[3] Athanassios. skodras, Charilaos Christopoulos and Touradj Ebrahimi, "The JPEG 2000 Still Image Compression Standard" IEEE signal processing magazine, 1053-5888, sep-2001.

[4] Bryan E. Usevitch, " A tutorial on Modern Lossy Wavelet Image Compression: Foundations of JPEG 2000", IEEE signal processing magazine, 1053-5888, sep-2001.

[5] Colm Mulcahy "Image Compression using the Harr wavelet transform", spelman science and Math Journal.

[6] J. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," IEEE Trans. Signal Processing, vol. 41, pp. 3445-3462, Dec 1993.

[7] R. A. DeVore, B. Jawerth and B. J. Lucier, " Imagecompression through wavelet transform coding" IEEE Trans. Informat. Theory, vol 38, pp. 719 - 746, Mar. 1992.

[8] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," IEEE Trans. Pattern Anal. Mach. Intell., vol 37, pp. 2091 - 2110, Dec. 1990.

[9] E. H. Adelson, E. Simoncelli, and R. Hingorani, “Orthogonal pyramid transforms for image coding,” Proc. SPIE, vol.845, Cambridge, MA, Oct. 1987, pp.50-58.

[10] Tang Xianghong Liu Yang “An Image Compressing Algorithm Based on Classified Blocks with BP Neural Networks” International Conference on Computer Science and Software Engineering, Date: 12-14 Dec. 2008 Volume: 4, On page(s): 819-822.

[11] Adnan Khashman, Kamil Dimililer “ Image compression using neural networks and haar wavelet” WSEAS Transactions on Signal Processing Volume 4 , Issue 5 , May 2008, Pages 330-339.

[12] Rafid Ahmed Khalil**,** “Digital Image Compression Enhancement Using Bipolar Backpropagation Neural Networks” Al-Rafidain Engineering Vol.15 No.4, 2007.

[13] Khashman, A. Dimililer, K. “Neural Networks Arbitration for Optimum DCT Image Compression” EUROCON, The International Conference on "Computer as a Tool" Sep. 2007 On page(s): 151-156..

[14] S. Anna Durai, and E. Anna Saro “Image Compression with Back-Propagation Neural Network using Cumulative Distribution Function” World Academy of Science, Engineering and Technology 2006.

[15] Hong Wang Ling Lu Da-Shun Que Xun Luo “Image compression based on wavelet transform and vector quantization” International Conference on Machine Learning and Cybernetics, 2002, 4-5 Nov. 2002 Volume: 4, on page(s): 1778- 1780

[16] Irina Perlieva, Viktor Pavliska, Marek Vajgl “Advanced Image Compression on the Basis of Fuzzy Transforms ” Submitted/to appear: IPMU 2008 (Malaga).

[17] L. Huang “Image compression based on fuzzy technique and wavelet transform” Conference Submissions ,Massey University at Albany, Auckland, New Zealand 2005.

[18] George E. Tsekouras, Mamalis Antonios, Christos Anagnostopoulos, Economou Dafni, and Damianos Gavalas “Image Compression Based on a Novel Fuzzy Learning Vector Quantization Algorithm”, University of the Aegean, Laboratory of Intelligent Multimedia and Virtual Reality, Greece.

[19] J. L. Su, Chen Yimin and Zhonghui Ouyang “An Image Compression Algorithm with Controllable Compression Rate" SpringerLink Saturday, July 07, 2007.