NEURAL NETWORK BASED IMAGE COMPRESSION WITH LIFTING SCHEME AND RLC

P.Srikala[1], Shaik Umar Faruq [2]

[1]Student (M.Tech), ECE Department, QUBA College of Engineering and Technology, AP, India,**psrikala86@gmail.com**
[2]Assoc Prof, ECE Department, QUBA College of Engineering and Technology, AP, India, **Faruq_sk3003@yahoo.co.in**

Abstract
Image compression is a process that helps in fast data transfer and effective memory utilization. In effect, the objective is to reduce data redundancy of the image while retaining high image quality. This paper proposes an approach for Wavelet based Image Compression using MLFF Neural Network with Error Back Propagation (EBP) training algorithm for second level approximation component and modified RLC is applied on second level Horizontal and Vertical components with threshold to discard insignificant coefficients. All other sub-bands (i.e. Detail components of 1st level and Diagonal component of 2nd level) that do not affect the quality of image (both subjective and objective) are neglected. With the proposed method in this paper CR (27.899), PSNR (70.16 dB) and minimum MSE (0.0063) of still image obtained are better when compared with SOFM, EZW and SPIHT.

Keywords: Image compression, wavelet, MLFFNN, EBP

---***---

1. INTRODUCTION

Data compression has become a necessity for saving bandwidth, power, storage space, etc. Thus it has turned out to be a present day craze as well as source of competition in the race for technology and research with so much manpower, time and money involved for its development. Out of the image compression techniques available, transform coding is the most preferred method. Since energy distribution after transform coding varies with each image, compression in the spatial domain is not an easy task. Images do however tend to compact their energy in the frequency domain making compression in the frequency domain much more effective. Transform coding is simply the compression of the images in the frequency domain. So transform based techniques like DWT, LWT, DCT, SVD, DWT-DCT, DWT-SVD, etc. have been extensively used [1].

Discrete Cosine Transform (DCT) has been the Transform of choice in image compression standard such as JPEG. DCT can be implemented in hardware. However, DCT suffer from blocking artifacts around sharp edges at low bit rate.

In general wavelets in recent years have gain widespread acceptance in signal processing and image compression in particular. Wavelet-based image coder comprises three major components: A Wavelet filter bank that decomposes the image into wavelet coefficients which are then quantized in a quantizer and finally an entropy encoder encodes these quantized coefficients into an output bit stream (compressed image). Although the interplay among these components is important and one has the freedom to choose each of these components from a pool of candidates, it is often the choice of the wavelet filter that is crucial in determining the ultimate performance of the coder. A wide variety of wavelet-based image compression schemes have been developed in recent years [2]. Most of these well known Image coding algorithms use novel quantization and encoding techniques to improve Coding Performance (PSNR). However, they all use a fixed wavelet filter built into the algorithm for coding and decoding all types of images, whether it is a natural, synthetic, medical, scanned or compound image. Wavelets have provided new class of powerful algorithm: They can be used for noise reduction, edge detection and compression. The usage of wavelets has superseded the use of DCTs for image compression in JPEG2000 image compression algorithm. Wavelet transforms are of different types. Some of them are given in [3].

Among learning algorithms, back-propagation algorithm is a widely used learning algorithm in artificial neural networks. The feed forward neural network architecture is capable of approximating most problems with high accuracy and generalization ability [12], [13]. This algorithm based on the error correction learning rule. Error propagation consists of two passes through different layers of the network, a forward pass and backward pass. In the forward pass input vector is applied to the sensory nodes of the network and its effect propagates through the network layer by layer. Finally a set of outputs is produced as the actual response of the network, during the forward pass the synaptic weight of the networks are all fixed. During the backward pass the synaptic weights are all adjusted in accordance with the error correction rule. The actual response of the network is subtracted from the desired response to produce an error signal. This error signal is then propagated back ward through the network against the direction of synaptic

conditions. The synaptic weights are adjusted to make the actual response of the network move closer to the desired response.

This paper is organized as follows: Importance and procedural steps of wavelet transform and Lifting scheme is explained in section II, multilayered feed forward neural network is explained in III, proposed method for compression system is explained in IV, Simulation results are presented in V, Conclusion and future scope is given in VI.

2. WAVELET TRANSFORM OF AN IMAGE

Wavelet transform is used to decompose an input signal into a series of successive lower resolution reference signals and their associated detail coefficients, which contains the information needed to reconstruct the reference signal at the next higher resolution level [17].

In discrete wavelet transform, an image signal can be analyzed by passing it through analysis filter bank followed by decimation operation. This analysis filter bank which consists of both low pass and high pass filters at each decomposition stage is commonly used in image compression. When signal passes through these filters, it is split into two bands namely low frequency band and high frequency band. The low pass filter, which corresponds to averaging operation, extracts the coarse information of the signal. The high pass filter, which corresponds to differencing operation, extracts the detail information of the signal. The output of the filtering operation is then decimated by two. A two dimensional transform can be accomplished by performing two separate one dimensional transform(Fig.1) First, the image is filtered along the X-dimension using low pass and high pass analysis filter bank and decimated by two. Low pass filtered coefficients are stored on the left part of the matrix and high pass filtered on the right. Because of decimation, the total size of transformed image is same as the original image.

It is then, followed by filtering the sub image along the Y-dimension and decimated by two. Finally, the image is split into four bands of Approximation component (LL1), Detail component (Horizontal (HL1), Vertical (LH1) and Diagonal (HH1)) through first level decomposition and second stage of filtering. Again the LL1 band is split into four bands viz LL2, HL2, LH2 and HH2 through second level decomposition. There exists several types of wavelets and for constructing biorthogonal wavelets, lifting scheme is a new method. Recently, biorthogonal wavelets constructed by the lifting scheme have been identified as very promising filters for lossless/lossy image compression applications.

Since the lifting scheme makes optimal use of similarities between the high and low pass filters, the computational complexity can be reduced by a factor of two compared with traditional fast wavelet transform algorithms.
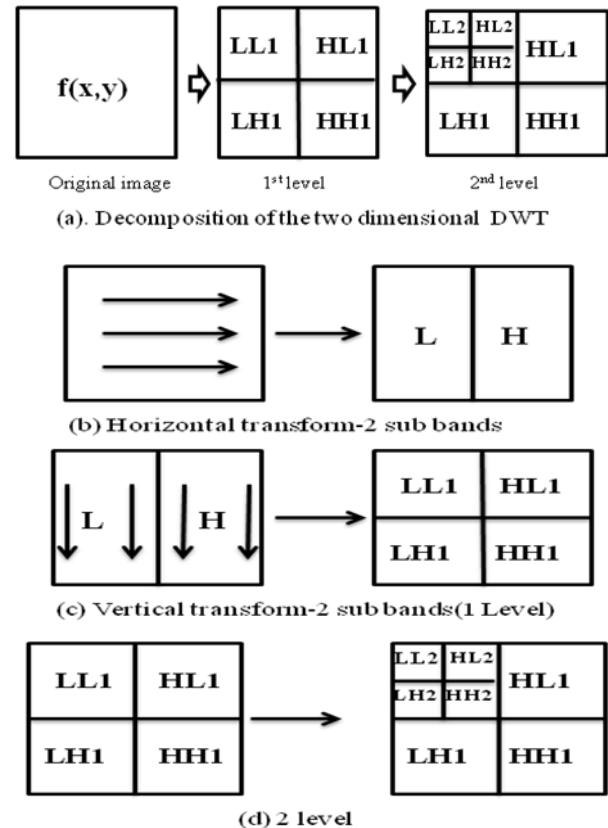

(a). Decomposition of the two dimensional DWT

(b) Horizontal transform-2 sub bands

(c) Vertical transform-2 sub bands(1 Level)

(d) 2 level

Fig-1: Decomposition of the two dimensional DWT

2.1 Lifting Scheme

Since the lifting scheme makes optimal use of similarities between the low and high pass filter, the computational complexity can be reduced by a factor of two compared with traditional fast wavelet transform algorithm.

With certain modifications, the corresponding wavelet transform can be calculated even with only integer addition and shift operations which makes the computation even faster. It can be used for lossless and lossy image compression because this transform is reversible. The Forward lifting scheme wavelet transform divides the data set being processed into an even half and odd half. Lifting scheme algorithms have the advantage that they do not require temporary arrays in the calculation steps, as is necessary for some versions of the Daubechies DB4 Wavelet algorithm. The Predict step calculates the wavelet function in the wavelet transform. This is a high Pass filter. The update step calculates the scaling function, which results in a smoother version of the data. This operation consists of three steps [4].

1) First, the input signal $s[n]$ is down sampled into the even position signal $se(n)$ and the odd position signal $so(n)$, then modifying these values using alternating prediction and updating steps. $se(n) = s[2n]$ and $so(n) = s[2n+1]$

2) A prediction step consists of predicting each odd sample as a linear combination of the even samples and subtracting it from the odd sample to form the prediction error.

3) An update step consists of updating the even samples by adding them to a linear combination of the prediction error to form the updated sequence. The prediction and update step may be calculated in several steps until the forward transform is reached.
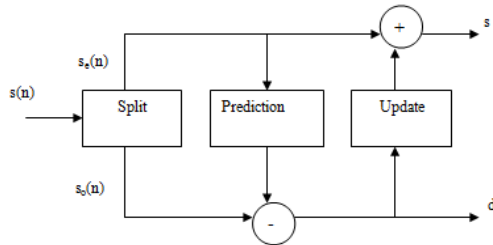

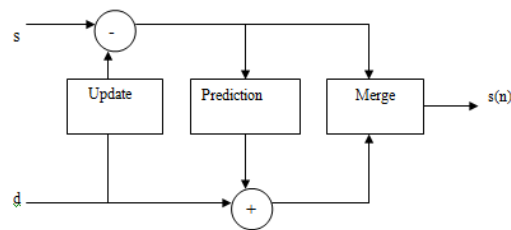**Fig-2**: Lifting scheme of forward wavelet transform.

Fig-3: Lifting scheme of inverse wavelet transform

3. MULTI LAYER FEED FORWARD NEURAL NETWORK:

Consider a feed forward neural network with a single hidden layer denoted by N-h-N, where N is the number of units in the input and output layers, and h is the number of units in the hidden layer. The input layer units are fully connected to the hidden layer units which are in turn fully connected to the output units. The output y, of the jth unit is given by

$$ Y_j = f \sum_{i=1}^{N} W_{ji} i + b_j \ \ldots\ldots (1) $$

$$ O_k = f \sum_{j=1}^{h} W_{kj} y_j + b_k \ \ldots\ldots (2) $$

Where, in eqn (1), Wji is the synaptic weight connecting the ith input node to the jth hidden layer, b, is the bias of the ith unit, N is the number of input nodes, f is the activation function, Y, is the output of the hidden layer. Analogously, eqn (2) describes the subsequent layer, where, Ok is the kth output in the second layer. The networks are trained using the variation of Back

propagation learning algorithm that minimizes the error between the network's output and the desired output. This error is given as follows.

$$ E = \sum_{k=1}^{N} \left(o_k - d_k \right) \ldots\ldots (3) $$

Where o and d are the present output and desired outputs of the kth unit of the output layer. For image compression, the number of units in the hidden layer h should be smaller than that in the input and output layers (i.e. h<N). The compressed image is the output of hidden layer neurons.

3.1 Image Compression Using MLNN

Every digital image is specified by the number of pixels associated with the image. Each pixel in a gray-level image is described by an intensity of the image at that point. An image that is 256 x 256, means that there are 65536 pixels (intensity points) in the image in a matrix form with 256 rows and 256 columns.

Grayscale image has different shade of gray which is commonly represented by 8 bits. So, there is 256 possible intensity values (shades of gray) for a grayscale image ranging from 0 to 255. Since 8 bits are used to represent each pixel, to represent an image which is of dimension [512 x 512, 512 x 512 x 8] = 2097152 bits are needed to represent the image.

MATLAB® has been used to implement the program. The well-known 'Lena' (bmp format is used here) grayscale image (512 x 512) has been used to demonstrate the technique. Each pixel in an image can be denoted as a coefficient, which represents the intensity of the image at that point. Then, the idea of compressing an image is to encode these coefficients with reduced bits and at the same time, retain the quality of the image to satisfactory limits. At the receiving end, these compressed images need to be again decoded or decompressed so that one can recover the original image. The quality of the received image can be tested by some standard error calculations. The mean of all the squared errors for all the pixels, called the MSE (Mean Square Error) can be used for this purpose. The higher the value of this MSE, lower the quality of the decompressed image.

The multi-layer feed-forward neural net has been used to compress images. The Lena image that has been used for compression purposes is a 512 x 512 image. This image can be broken into blocks of size 8 x 8. There will then be 64 pixels per block. Totally, there will be [64x 64] blocks. The 64 pixels in each block then becomes the input vector to the neural net. If one can reduce the number of dimensions in the hidden layer (number of hidden neurons) to be much less than the number of dimensions in the input layer, then there will be a reduction in the number of coefficients during the transition from the input to

the hidden layer. An input layer with 64 dimensions and a hidden layer with 8 (any number less than 64) dimensions, for example, means that the 64 pixels of the image (8 x 8) block which is applied to the input layer has been transformed into 8 coefficients in the hidden layer. Then, one could again use an output layer which has 64 dimensions to recover the original 64 pixels. The basic idea here is to learn the identity mapping or rather associative mapping which means the output of the neural net is the same as its input.

Thus, with a 64 dimensional input layer, an 8 dimensional hidden layer, and a 64 dimensional output layer – the neural network has been used for image compression. In this image compression technique, the compression is achieved by training a neural network with the image and then using the weights and the coefficients from the hidden layer as the data to recreate the image. This will be very clear with an example.

If each pixel is encoded using 8 bits, then the total number of bits to be transmitted without compression is [512 x 512 x 8] for a [512 x 512] pixel image. A [512 x512] pixel image is split into [4 x 4] or [8 x 8] or [16 x 16] pixel sub-images. The normalized pixel value of the sub-image is the input to the nodes. 64 input layers (in case of [8 x 8] sub-image size) are taken with 1 pixel input to each layer. The three-layered back propagation-learning network has been trained with each sub image.

The number of neurons in the hidden layer will be taken according to the desired compression. Here, we have taken 8 hidden layers. The number of neurons in the output layer will be the same as that in the input layer (64 in our case). The input layer and output layer are fully connected to the hidden layer.

The weights of synapses connecting input neurons and hidden neurons and weights of synapses connecting hidden neurons and output neurons are initialized to small random values from say – 1 to +1.

Only the weights between the hidden layer and the output layer are required for reconstruction. So, the numbers of weights are [64 x 8] and number of bits used to represent them are [64 x 8 x 8]. The input layer uses sigmoid function.

The hidden layer units evaluate the output using the sigmoid function. The output layer neuron evaluates the output using linear activation function. As the image is split into [8 x 8] pixel blocks, the total number of blocks becomes [64 x 64] and for each sub-image, the number of coefficients out of the hidden layer is 8. So, the total numbers of coefficients are [64 x64 x 8]. Thus, total number of bits required to represent the coefficients are [64 x 64 x 8 x 8].

Total number of bits to be transmitted without compression, Tb = [512 x 512 x 8]. After training the network, number of bits to be transmitted, Cb = [(64 x 64 x 8 x 8) + (8 x 64 x 8)].

Therefore compression achieved is:

Percentage of Compression = [(1 – Cb/Tb) x 100%].
With the method adopted in this paper LL2 component obtained after applying wavelet is compressed using multi layered feed forward neural network. Consider   network has N units in the input and output layers, h1 units in the hidden layers.

Fig.4 depicts the system during the training phase, Network is trained to compress and decompress the image.

Let X be the input image of Network and Y is its output. The compression ratio is defined by

$$CR = \frac{N}{Z1 + Z2} \ldots\ldots\ldots (4)$$

Where N is the dimension of the image, Z1 and Z2 are the number of hidden units in Network. The dimension of the compressed image C is Z1+Z2.

As in Fig.5, the Coder compresses the input of Network, and Decoder decompresses the output of the hidden layer of Network.

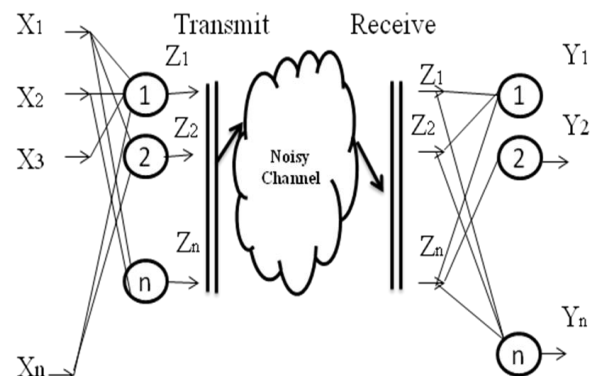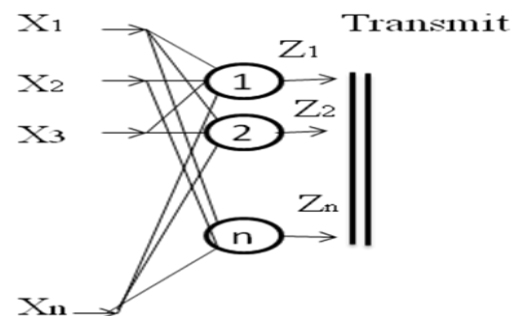**Fig-4:** Neural system
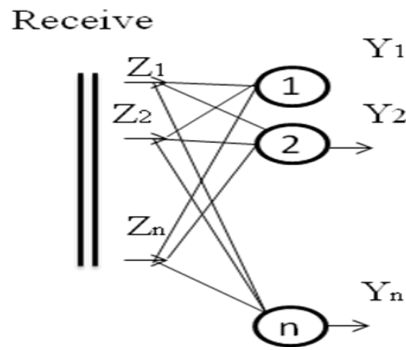

Fig-5: The image compressor system.

Fig-6: The image decompressor/decoder system.

4. PROPOSED METHOD:

Image compression can be achieved by decomposing the image using Biorthogonal wavelet (2nd level), Processing LL2 component by using MLFF Neural Network with Error Back Propagation (EBP) training algorithm and applying RLC on LH2 and HL2 components with hard threshold to discard insignificant coefficients. This is clearly explained in below Fig.7 and algorithm explains the encoding and decoding process.
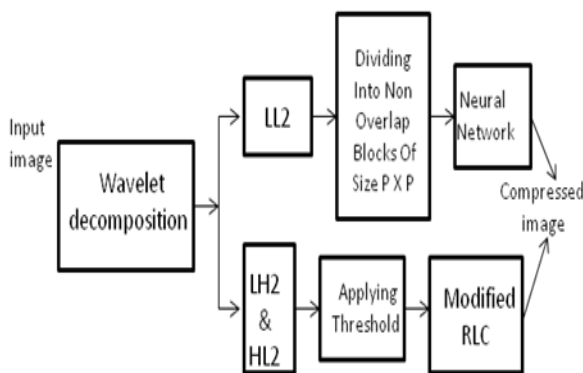

Fig-7: Proposed compression system

4.1 Algorithm for proposed method:

1. Reading the image from the data base.
2. Applying wavelet on image.
3. Discarding sub bands LH1, HL1, and HH1 of first level and HH2 of second level. Dividing LL2-sub band into non overlapping sub blocks of size P x P.
4. Applying Threshold to LH2 & HL2-sub bands to discard insignificant coefficients.
5. Encode the threshold coefficients using Modified run length coding.
6. Sub blocks of LL2-sub band is given as input to neural network for training.
7. Weight matrix between hidden and output layers, hidden layer output and Modified run length encoded sequence are meant for storage or transmission.

4.2 Algorithm for Run Length Coding:

1) Encoding:
1. Read the input vector a (i) and convert in to a single row.
2. Separate the non zero values from input vector a(i), and all non zero values are placed in b(i) .
3. Replace positions of non zero values with '1' in a (i).
4. Now apply run length encoding for a (i).
5. b (i) and second part of encoded a (i) , has to be stored or transmitted**.**

2) Decoding:
1. Reading the encoded a (i) and b (i).
2. Generate sequence (010101/101010) of size a(i) and concatenate with a(i)
3. Decode vector a(i) and
4. Replace positions of all 1's in vector a(i) with non zero elements from b (i) in the order and reorder vector a (i).

4.3 Procedural Steps for Neural Network Application:

1. Dividing the original image into m blocks by l pixels and reshaping each one into column vector;
2. Arranging all column vectors in a matrix;
3. Choosing a suitable training algorithm and defining the training parameters: the number of iterations, the number of hidden neurons and the initial conditions;
4. Simulating the network using input data, result matrices and an initial error value;
5. Rebuilding the compressed image; finally,
6. Terminate the calculation if error is smaller than the threshold decided before [6].

5. SIMULATION RESULTS AND DISCUSSIONS

In this proposed method multilayer feed forward neural network (three layers) with error back propagation training algorithm having, different types of images of size 512 x 512 pixels are used and algorithm is implemented using MATLAB. The PSNR (peak signal to noise ratio) based on MSE (mean square error) is used as a measure of "quality", MSE and PSNR are calculated by the following relations:

$$MSE = \frac{1}{MXN} \sum_{i=1}^{M} \sum_{j=1}^{N} \left(x_{i,j} - y_{i,j}\right)^2 .... (5)$$

$$PSNR = 10\log_{10}\left[\frac{(255)^2}{MSE}\right] .... (6)$$

Where M x N is the image size, $x_{i,j}$ is the input image and $y_{i,j}$ is the reconstructed image. MSE and PSNR are inversely proportional to each other and high value of PSNR guarantees better image quality. Mean square error (MSE), visual quality, Compression ratio and PSNR of image is calculated and compared with existing lossless and lossy compression

methods in table 1. Here compression ratio (**27.89**) with good visual quality, PSNR (**70.16dB**) and MSE (**0.0063**) is obtained. In SOFM, compression ratio, reconstructed image quality is poor. SPIHT and embedded zero wavelet (EZW) [5] gives an acceptable visual quality but with poor compression ratio, and computationally expensive.

Table-1: Performance Comparison of Different Methods, For Lena Image

Technique	MSE	PSNR	CR
Proposed method	**0.0063**	**70.1602**	**27.8995**
SPIHT	25.33	34.09	5.95
EZW	46.11	31.49	3.07
SOFM	73.72	29.45	8.92

Table-2: MSE, PSNR, CR Values For Lena Using Bior6.8

Method	Threshold		MSE		PSNR		CR	
	HL	LH						
Only Runlength	-----	----	0		Infinity		0.5610	
			Level1	Level2	Level1	Level2	Level1	Level2
Only DWT	0.08	0.08	2.317e-4	7.499e-4	84.48	79.38	3.67	11.12
Only LWT	0.08	0.08	4.90e-04	0.0063	81.22	70.16	3.64	11.69
DWT & Run length	0.08	0.08	2.31e-04	7.49e-04	84.48	79.38	3.61	9.90
LWT & Run length	0.08	0.08	4.90e-04	7.49e-04	81.22	79.38	3.53	9.90
LWT , Run length & Neural Network	0.08	0.08	4.94e-04	0.0063	81.18	70.16	20.71	27.89
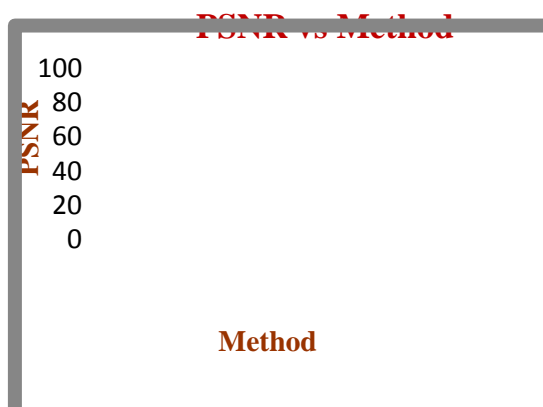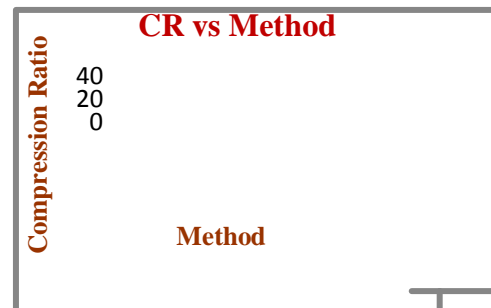
Chart-2: CR vs Metod

Input Image

Reconstructed Image

Error Image

Fig- 9: Input, Reconstructed and Error images

Chart-1: PSNR vs Meto d

CONCLUSION AND FUTURE SCOPE

Neural Network Based Image Compression with Lifting Scheme and RLC is done with MLFFNN (single hidden layer) with EBP training algorithm for LL2 component. Modified run length coding is applied on LH2 and HL2 components with threshold to discard insignificant coefficients and neglecting all other sub-bands (HH2, LH1, HL1 and HH1). Maximum compression ratio of **27.89**, PSNR of **70.16 dB**, minimum MSE of **0.0063** with threshold of **0.08 & 0.08** for both **LH2** and **HL2** sub bands to image lena of size 512x512 compared to SOFM, EZW and SPIHT [5]. It is found that the proposed method is superior to existing compression methods. This can be further enhanced with Radial Basis Function (RBF) networks and bidirectional associative memory networks because training time will be reduced by many folds.

REFERENCES

[1]   D.P. Dutta, S.D. Choudhury, "Digital Image Compression Using" 2009 International Conference on Advances in Computing, Control, and Telecommunication Technologies. ACT 09'. 28-29 Dec 2009.

[2]   J.Villasenor, B. Belzer, J. Liao, "Wavelet Filter Evaluation for Image Compression," *IEEE Transactions on Image Processing*, Vol. 2, pp. 1053-1060, August 1995.

[3]   Rohit Arora et al, "An Algorithm for Image Compression Using 2D Wavelet Transform" International Journal of Engineering Science and Technology (IJEST), Vol. 3 No. 4 Apr 2011.

[4]   Chesta Jain and Kapil Jain," Performance Analysis of Integer Wavelet Transform for Image Compression**"** *IEEE Transactions on Image Processing,* 978-1-4244-8679-3/11/$26.00 ©2011 IEEE

[5]   S.P. Raja , Dr. A. Suruliandi " Analysis Of Efficient Wavelet based Image Compression Techniques" 2010 Second International conference on Computing, Communication and Networking Technologies.

[6]   Weiwei Xiao, Haiyan LiuCollege of ScienceNorth China University of TechnologyBeijing, P. R. China" Using Wavelet Networks in Image Compression" 2011 Seventh International Conference on Natural Computation.

[7]   James S. Walker., *A Primer on Wavelets and Their Scientific Applications, Second edition,* Taylor & Francis Group,   LLC, Beijing, Jun.2008.

[8]   V. Mohan Y. Venkataramani "Compression of Iris images using DTCNN based Wavelet Decomposition and Directional   Filter Bank Analysis" proceedings of IEEE conference 2011.

[9]   Michel Misiti, Georges Oppenheim, Jean-Michel Poggi,Yves Misiti, *Wavelet Toolbox™ User's Guide, Second Edition*, Minor revision for Version 4.4.1 (Release 2009b),Online   only, Beijing, Sep.2009.

[10]   Peter L Venetianer, Tomas Roska, "Image compression by Cellular Neural Networks", IEEE Trans. On Circuits and systems –I, Vol.45, No.3, March 1998.

[11]   N. Senthil kumaran, *Member IACSIT* and Dr. J. Suguna "Neural Network Technique for Lossless Image Compression Using X-Ray Images" International Journal of Computer and Electrical Engineering, Vol. 3, No. 2, April, 2011

[12]   H. Simon, Neural Networks and Learning Machines. 3rd ed. Beijing, China: China-Machine, pp. 124-156, 2009

[13]   Dong Changhong, Neural Networks and Applications. 2rd ed. Beijing, China: National Defense Industry, pp. 14-120, 2009.

[14]   Hadi Veisi, Mansour Jamzad" A Complexity-Based Approach in Image Compression using Neural Networks" International Journal of Information and Communication Engineering 5:2 2009.

[15]   Xiulian Peng, Jizheng Xu, *Member, IEEE*, and Feng Wu, *Senior Member, IEEE"* Directional Filtering Transform forImage/Intra-Frame Compression*"* IEEE Transactions On Image Processing, Vol. 19, No. 11, November 2010.

[16]   Khashman A. and Dimililer K., Image Compression using Neural Networks and Haar Wavelet, *WSEAS Trans Signal*

   rocessing 4 (5), pp. 330-339, May 2008.

[17]   *Digital Image Processing* by Esak kirajan S, *Veerakumar* T, Jayaraman S, MGH International.

BIOGRAPHIES:

P. Srikala has Received B.tech in Electronics and Communication Engineering (ECE) from JNTU, Hyderabad, India in 2007. She is persuing her M.Tech in Quba college of engineering and technology, JNTUA, Anantapur.

Shaik Umar Faruq is working as a Associate Professor in QUBA College of engineering and Technology, Nellore. He has 9 years teaching experience both at UG and PG level and 1 year industrial experience.