A Multilayered Perceptron Reconfigurable ANN (Artificial Neural Networks) Architecture for Image Compression & Decompression.

Mr.Pramod.V.Patil[1]
M. Tech (VLSI Design and Embedded system),
H.I.T Nidasoshi,Belgaum
Patil.pramod080@gmail.com

Dr.M.N.Shanmukha swamy.[2]
Professor, Dept of ECE,
SJCE,Mysore,India

Dr. Cyril Prasanna Raj P.[3]
Professor Dept. of EEE
MSRSAS Bangalore ,India

Abstract - There are number trials of using neural networks as signal processing tools for image compression. In this paper, a direct solution method is used for image compression using the neural networks. An experience of using multilayer perceptron for image compression is presented. The multilayer perceptron is used for transform coding of the image. The network is trained for different number of hidden neurons with direct impact to compress ratio. It is experimented with different images that have been segmented in the blocks of various sizes for compression process. Reconstructed image is compared with original image using signal-to-noise ratio and number of bits per pixel. The results show the possibility of using multilayer perceptrons for image compression.

Keywords - Image compression, neural networks, Multilayer perceptron, Training algorithm, NMSE, SNR.

I. INTRODUCTION

Image compression is playing key role in the development of various multimedia computer services and telecommunication applications. As image needs a huge amount of data to store it, there is pressing need to limit image data volume for fast transport along communication links. The goal of image compression techniques is to remove redundancy present in data in a way that enables image reconstruction. Statistical properties of the image are used in design an appropriate compression technique. The strong correlation between image data items enables reduction in the data contents without significant quality degradation. There are numerous lossy and lossless image compression techniques. Considering X-ray images or ECG data where each bit of information is essential a lossless compression must be used. On the other hand, for the still digital image or video, which have been already lossy digitalized, a lossy compression is preferred. There are numerous lossy compression techniques. Very effective techniques are transform based techniques, particularly cosine transform based technique that showed excellent results. Contrary traditional techniques for image compression, it is worth to discuss some of the recent techniques that may be used for data compression.

Artificial Neural Networks (ANN) has been used for solving many problems, special in cases where the results are very difficult to achieve by traditional analytical methods. There have already been a number of papers published applying ANN to image compression [1], [2], [3], [4], [5]. It is important to emphasize, although there is no sign that neural networks can take over the existing techniques, research on neural networks for image compression are still making some advances. Possibly in the future this could have a great impact to the development of new technologies and algorithms in this area. The main goal of this paper is to investigate which neural network structure accomplished with different learning algorithms give the best results compared with standard transform coding techniques. The various feed forward neural networks with back propagation algorithms were directly applied to image compression coding. In Section 2 some principles of transform coding using neural networks are discussed. Section 3 analyses experimental results obtained with different neural networks structures and images. The conclusion and summary of present research are presented in Section 4.

II. TRANSFORM CODING WITH MULTILAYER PERCEPTRON

Generally, transform coding needs the image to be subdivided into non-overlapping blocks of n x n pixels each. Such block represents N-dimensional vector x, $N = n \times n$, in N-dimensional space. Transformation process maps this set of vectors into another M-dimensional space, where $M<N$ [6]. The inverse transformation need to reconstruct original image with minimum of distortions. In the case of linear transformation coding M x N dimensional matrix W is performed on each block with M rows (w), so compressed vector y can be calculated as

The compression-decompression by comparing the resulting pi) (original) pixels, using normal error (NMSE) [4]. NMSE for k b follows:

$$\mathbf{y} = \mathbf{W}\mathbf{x} \qquad (1)$$

The reconstructed vector can be calculated as 95

$$\mathbf{o} = \mathbf{W}^{T}\mathbf{y} \qquad (2)$$

The main task here is to find matrix W with specified M rows of the eigenvectors so that deviation between y and o is minimized and expressed through mean square error (MSE).

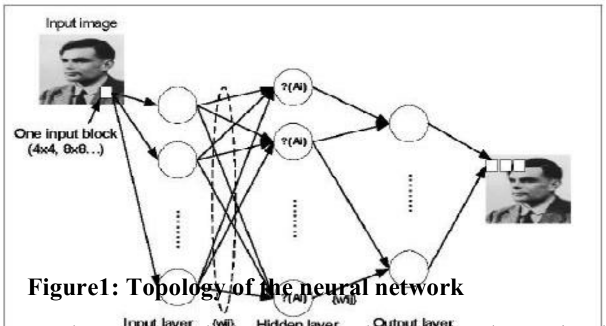


Figure1: Topology of the neural network

A two-layer network is considered as it is shown in the Fig.1.Each neuron of the input layer corresponds to a pixel of the input block. The neurons of the hidden layer are used as the components on which the input blocks will be projected. This operation can be formulated as

$$y_j = \varphi\left(\sum_{i=1}^{N} w_{ji} x_i + b_j\right) \qquad 1<=j<=M$$

Where p(.) is nonlinear activate sigmoid function, and bj is bias hidden neuron [6]. The neurons correspond to the decompression block. The compressed image consist of the input blocks on the principal components (wji, xi), and weight vectors permit According to the above, number specifies the compression ratio. The decompression is performing toward output layer. The identity is used for the output layer, so for decompression we have

$$o_i = \sum_{j=1}^{M} w'_{ij} y_j \qquad 1<=i<=N$$

$$NMSE = \frac{\sum_{i=1}^{k}\sum_{j=1}^{N}(\bar{I}_{ij} - I_{ij})^2}{\sum_{i=1}^{k}\sum_{j=1}^{N} I^2_{ij}}$$

Where Iij is pixel intensity value of pixels in the reconstructed image, while Iij is pixel intensity value of pixels in the original image. The original and reconstructed images are split into k input vectors:

$$x_i = \{I_{i1}, I_{i2}, I_{i3}, \ldots I_{iN}\}$$

As an alternative to MSE as indication of decompressed images quality signal-to-noise-ratio (SNR) is introduced **:**

$$SNR = 10\log_{10} \frac{\sum_{i=1}^{k}\sum_{j=1}^{N} I_{ij}^{\;2}}{\sum_{i=1}^{k}\sum_{j=1}^{N}(\bar{I}_{ij} - I_{ij})^2} (dB)$$

The important measure of achieved compression is compression rate as average number of bits per pixel for the compressed image. The compression rate can be expressed as where t, is the number of bits associated to the ith pixel in the block, and p is the number of pixels in the block.

$$CR = \frac{\sum_{i=1}^{M} t_i}{p} (bpp)$$

IV. Using Multi-Layered Networks

It is well known in neural network science that when classes cannot be separated by a hyper plane, the single layer perceptron is not appropriate [SI]. Multi-layer perceptrons overcome many of the limitations of single-- layer perceptrons. Multi-layer perceptrons are feed forward nets with one or more layers of nodes between the input and output nodes. The capabilities of multi-layer perceptrons stem from the non-linearity's used within the nodes. Indeed, if the nodes of the multi-layer neural network were linear, the multi-layer neural network would perform identically as a single-layer one. In this work, it is understood that multi-layer networks have only non-linear nodes. **A** two-layer perceptron can form any, possibly unbounded, convex region in the space spanned by the inputs. **A** three-layer perceptron can form arbitrarily complex decision regions and can separate the meshed classes[S]. Kolmogorov in [9] showed that a three layer perceptron with ***N(*** 2 ***N+*** 1) nodes using continuously increasing non-linearity can compute any continuous function of N variables. **A** three-layer perceptron could thus be used to create any continuous likelihood function required in a classifier. The above discussion on the classification and approximation capabilities of the multi-layer neural network is based on the fact that all the layers are trained simultaneously. It is important to emphasize here that training each layer separately, as was done in [4], does not share the proven capabilities of the standard multilayer neural network. The operation of the 2-layer neural network of Figure 1 can be viewed as a coder and decoder, each of which is a single layer perceptron. Based on the above, one can conclude that the network in Figure 1 gives, in general, a sub-optimal solution. To correct this deficiency it is proposed here that both the coder and the decoder be implemented by a multi-layer neural network. As a start, 4-layer neural network (2-layers for each) will be used as shown in Figure 2.

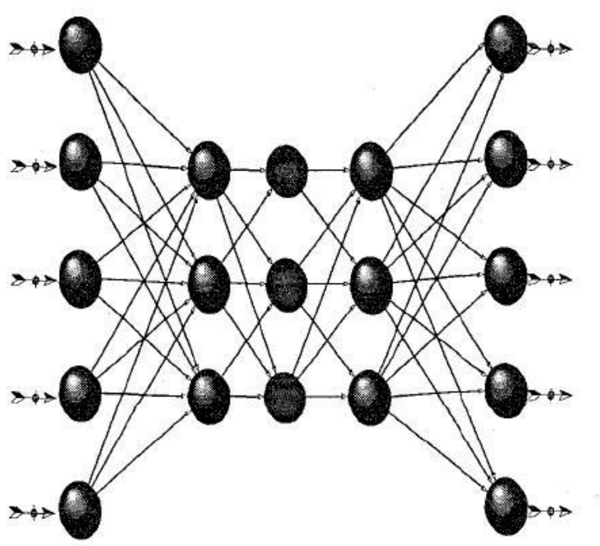


Figure 2. The proposed 4-layers Feed-Forward Neural Net used to compress images.

5. RESULTS

1. Original Image taken for compression and decompression

Figure 3: Original image

2. Part of the image (64*64resolution) captured

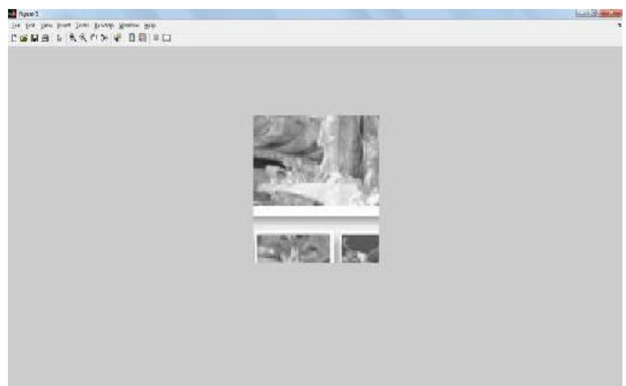


Figure 4: part of the original image

3. Part of the image after Decompression

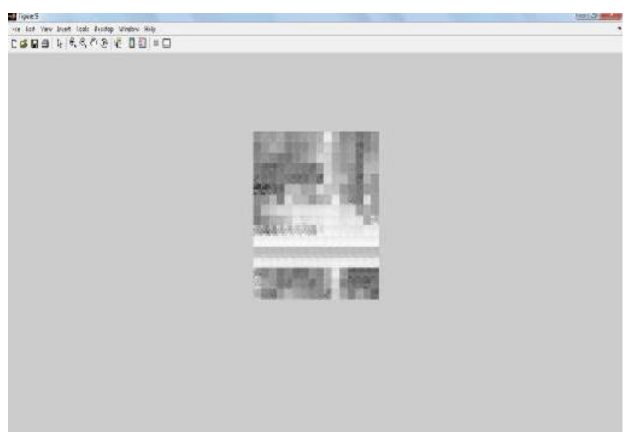


Figure 5 : Part of the image after Decompression

4. Performance Graph of the network after training

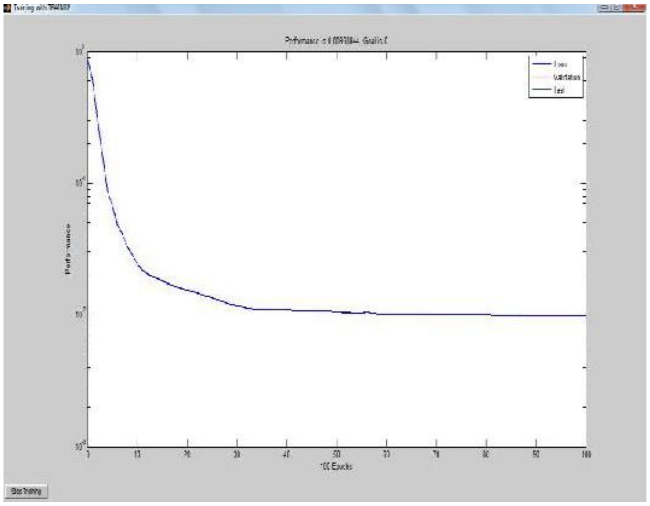


Figure 6: Performance Graph of the network after training

5. Image Maximum Error(IME)

After the training and calculating the maximum Error which is the difference of original image and compressed image is

IME =152

6. Mean Square Error (MSE)

Mean square Error of the image after calculation compression and decompression coefficients of the image is

MSE =630.6667

7. Peak Signal to Noise Ratio

Peak Signal to noise ratio is calculated after the image compression and decompression is

PSNR =20.1328

.

8.Schematic of the top module of ANN architecture in Xilinx13.1

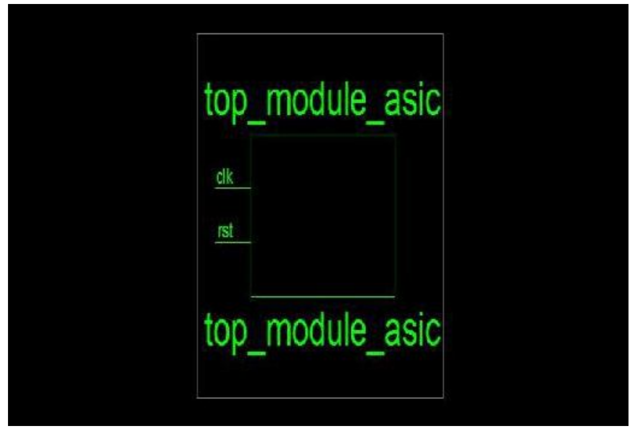


Figure 7: Schematic of the top module of ANN

9.Schematic of the Compression module of ANN architecture in Xilinx13.1

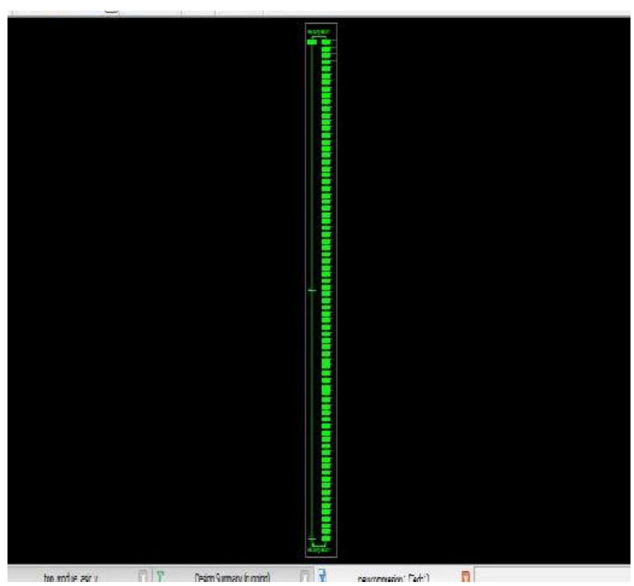


Figure 8: Schematic of the Compression module

8.Schematic of the TanSig function module of ANN architecture in Xilinx13.1

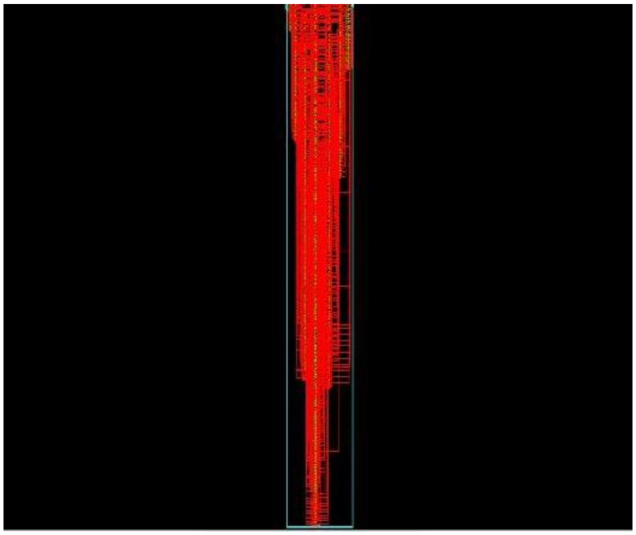
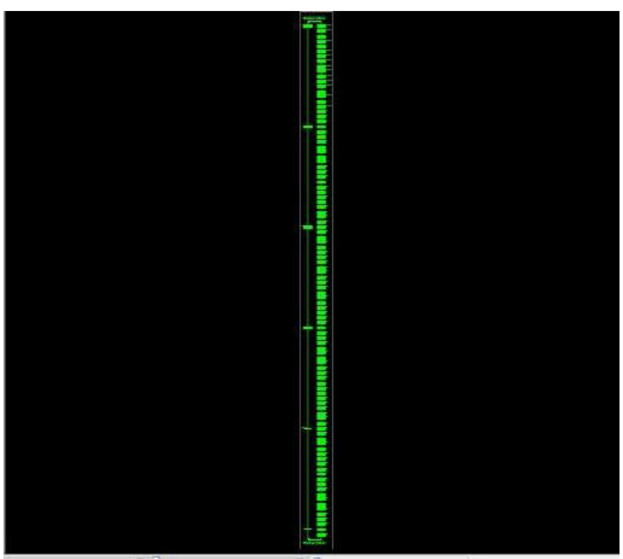


Figure 9: Schematic of the TanSig function module

8.Schematic of the Decompression module of ANN architecture in Xilinx13.1



. **Figure 10: Schematic of the TanSig function module**

6. CONCLUSIONS

The aim of this paper was to present some experience in using neural networks for image compression. The multilayer perceptron was used for direct image compression. The main results showed good quality of reconstructed images with relative low compression rates, but it can't be used instead some standard compression codec, like JPEG. According to the obtained results it can be concluded that it is worth to continue developing new algorithms for image compression using neural networks **.**

REFERENCES.

[1] R. D. Dony, and **S.** Haykin, "Neural Network Approaches to Image Compression," ***Proceedings of the IEEE,*** **Vol. 83,** No. **2,** pp. **288-303,** Feb. **1995.**

[2] G. W. Cottrell and P. Munro, "Principle Components Analysis of Images via Back Propagation," in ***SPIE*** Vol. **1001** ***Visual Communication And Image Processing*** **'88** pp. **1070-1077, 1988.**

[3] Sonehara, N., Kawato, M,. Miyake, S., and Nakane, K., "Image Data Compression using Neural Network Model," ***Proc.*** IJCNN, **1989,** Washington DC, pp. **11-35-11-41.**

[4] Namphol, A., Chin, ***S.*** H., and Arozullah, M., "Image Compression with Hierarchical Neural Network," ***IEEE Tran. Aerospace and Electronic systems,*** **Vol. 32,** No. **1,** pp.

[5] **L.** E. Russo and C. E. Real, "Image Compression using Outer Product Neural Network," ***Proc. IEEE Int.*** *ConfAcoust.* ***Speech, and Signal Process.*** **'92,** pp. **I1 377-380,** SanFrancisco, CA, Mar. **1992.**

[6] Carrato, **S.,** Premoli, A., and Sicuranza, G. L., "Linear and nonlinear Neural Networks for Image Compression,"***Proc.*** **1991** ***Int.*** *Conf on* ***Digital Signal Processing,*** Sept. **1991,**

[7] Kono,R., **M.** Arai, and H. Imai, "Image compression using neural network with learning capability of variable function of a neural unit," SPIE vol. **1360,** ***Visual Communication andImage Proc.*** **'90,** pp. **69-75, 1990.**

[8] A. Namphol, S. Chin, M. Arozullah, "Image Compression with Hierarchical Neural Network", IEEE Trans. Aerospace and Electronic Systems, Vol. 32, No. 1, pp. 326-338, Jan. 1996.

[9] L.E. Russo, C. E. Real, "Image Compression Using Outer Product Neural Network", Proc. IEEE Int. Conf Acoust. Speech and Signal Process, Vol. 2 pp. 377-3 80, San Francisco, CA, 1992.

[10] J. Jiang, "Image Compression with Neural Networks - A Survey", Signal Processing. Image Communication, Vol. 14, pp. 737- 760, 1999.

[11] S.Costa, S. Fiori, "Image compression using principal component neural networks", Image and Vision Computing 19, pp. 649-668, 2001.

[12] R. D. Dony, S. Haykin, "Neural Network Approaches to Image Compression", Proceedings ofIEEE, Vol. 83, No. 2, pp. 288-303, February 1995.

[13] S. Haykin, Neural Networks, A Comprehensive Foundation, Prentice Hill, Upper Saddle River, New Jersey, 1999.

[14] Jianxun MI, De-Shuang Huang, "Image Compression UsingPrincipal Component Neural Network", 8thInternational Conference on Control, Automation, Robotics and Vision, Vol. 1, pp. 698-701, 6-9 Dec. 2004.