Image Compression with Neural Networks Using Complexity Level of Images

Hadi Veisi and Mansour Jamzad

Sharif University of Technology
Department of Computer Engineering, Tehran, Iran
veisi@ce.sharif.edu, jamzad@sharif.edu

Abstract

This paper presents a complexity-based image compression method using neural networks. In this method, different multi-layer perceptron ANNs are used as compressor and de-compressor. Each image is divided into blocks, complexity of each block is computed using complexity measure methods and one network is selected for each block according to its complexity value. Three complexity measure methods, called Entropy, Activity and Pattern-based are used to determine the level of complexity in image blocks and their ability are evaluated and compared together. Selection of a network for each image block is based on its complexity value or the Best-SNR criterion. Best-SNR chooses one of the trained networks such that it results best SNR in compressing a block of input image. In our evaluations, best results, with PSNR criterion, are obtained when overlapping of blocks is allowed and choosing the networks in compressor is based on the Best-SNR criterion. In this case, the results demonstrate superiority of our method comparing with previous similar works and that of JPEG standard coding.

Key words: *Image compression, image complexity, multi-layer perceptron, neural network, back-propagation, JPEG, PSNR.*

1. Introduction

Despite all advances, image compression is still an important subject in many areas such as communication, data storage and computation. In addition to standards and classical coding methods, artificial intelligent technologies such as neural networks are being developed for image compression. Learning capabilities, noise suppression, transform extraction, parallelism and optimized approximations are some main reasons that encourage researchers to employ artificial neural networks in many applications. Although there are no significant works on neural networks that can take over the existing image compression technology but there are some admissible attempts. Many types of networks such as MLP[1] [1-7], Hopfield[8], SOM[2] [9], LVQ[3] [10] and PCA[4] [11] are experimented as a compression method. Among these methods, the MLP network which usually uses back-propagation training algorithm provides simple and effective structures. It has been more considered in comparison with other ANN structures. The compression of images by Back-Propagation Neural Networks (BPNN) is investigated by many researchers. One of the first tries in using this approach was done in [1], in which the authors proposed a three layer BPNN for compressing images. In their method original image is divided into blocks and fed to input neurons, compressed blocks are found at the output of the hidden layer and the de-compressed blocks are restored in the neurons of output layer. The simulation results of this method showed that this network provided a poor image quality even for trained images in 4:1 compression ratio [1]. As pointed out in [6], none of the results in using single network are so good as the result that could be achieved by taking average of image blocks and using their values as the indicator of blocks!. In [7] a method is proposed for improving the performance of simple BPNN structure using hierarchical neural networks which extended BPNN by adding two more hidden layers to it. Although this extension exploits the correlation between blocks in addition to the correlation between pixels, its improvement is not so considerable.

Adaptive methods use another approach to compress/de-compress (CODEC) the image blocks. In this approach various networks are used for compress/decompress different image blocks regarding to the complexity of blocks. This approach provides admissible results in compression with neural networks. In [3] it is suggested to cluster image blocks into four classes based on activity of image blocks. Another adaptive approach which proposed in [6] classified image blocks into six classes based on their orientation. Also, an extension of this approach is given in [4] in which blocks are classified into nine predefined orientations for reducing edge degradation. After selecting related network in compressor, each pixel in a block is subtracted from the mean value of that block. These methods have yielded significant improvement over basic BPNN.

In this paper we have used basic neural network-based algorithm for compressing images. Then an adaptive approach based on three complexity methods named Entropy, Activity and Pattern-based is presented. In evaluation, in addition to using complexity criterion to select the appropriate network for compressing an incoming image block, Best-SNR method is also used

[1] - Multi-Layer Perceptron
[2] - Self-Organizing Map
[3] - Learning Vector Quantization
[4] - Principal Component Analysis

to select one of the trained networks. It selects a network that gives best SNR for that image block among all networks. Also, overlapping of image blocks is used in order to eliminate the chess-board effect in de-compressed image.

The reminder of this paper is organized as follows. In section 2, we discuss using of multi-layer perceptron neural network with back-propagation training algorithm and its adaptive approach in image compression. Section 3 describes the complexity measurement methods used in this paper. In section 4, the experimental results of our implementations are discussed and finally in section 5 we conclude this research and give a summary on it.

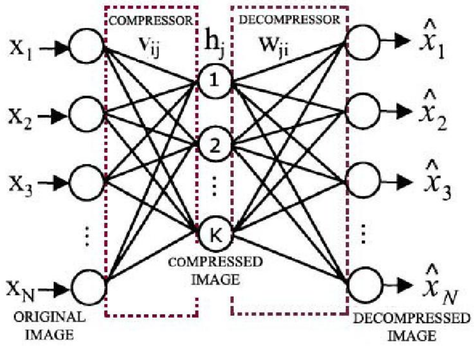
2. Multi-Layer Perceptron Neural Network for image compression

Multi-Layer Perceptron (MLP) neural networks with back-propagation training algorithm can directly be applied to image compression. The simplest neural network structure for this purpose is illustrated in Figure-1 (a). In this three layers network, both input and output layers are fully connected to the hidden layer and have the same number of neurons: N . Compression can be achieved by allowing the value of the number of neurons at the hidden layer: K, to be less than that of number of neurons at both input and output layers ($K \le N$). As in most compression methods, the input image is divided into blocks of size, for example 8×8 , 4×4 or 16×16 pixels. These block sizes determine the number of neurons in the input/output layers which convert to a column vector and fed to the input layer of network; one neuron per pixel. We call this structure as basic neural network and train it using back-propagation algorithm. Training blocks are converted into vectors and then normalized from their gray-level range into [0,1]. In accordance with the structure of neural network shown in Figure-1, the operation for adjusting weights for compressing and de-compressing can be described as the following equations.

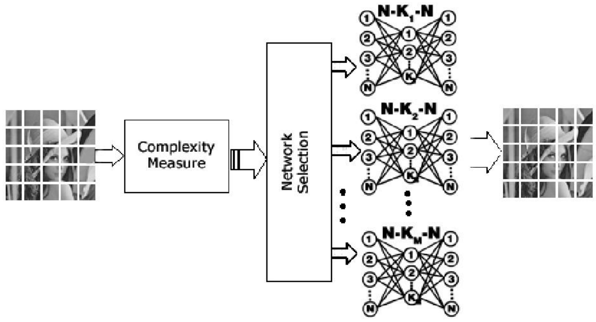
$$H_j^{in} = \sum_{i=1}^{N} V_{ij} X_i \, , \quad h_j = f(H_j^{in}); \; 1 \le j \le K \tag{1}$$

$$\hat{X}_i^{in} = \sum_{j=1}^{K} W_{ji} h_j \, , \quad \hat{X}_i = g(\hat{X}_i^{in}) \, ; \; 1 \le i \le N \tag{2}$$

Where f and g are the activation functions that can be linear or nonlinear. V_{ij} and W_{ji} represent the weights of compressor and de-compressor, respectively. The extracted $N \times K$ transform matrix in compressor and $K \times N$ in de-compressor of linear neural network are in direction of PCA transform. Although this transform provides optimum solution for linear narrow channel type of image compression and minimizes mean square error between original and reconstructed image, but unfortunately, it is a data-dependent transform. Using linear and non-linear activation functions in this network results linear and non-linear PCA, respectively.



(a)



(b)

Figure 1: (a) Basic neural network structure, (b) adaptive structure, for image compression

Poor compression performance of the basic structure and the extracted transforms makes it useless in practice. Adaptive approach [2-6] is used to improve the ability of this method. As shown in Figure-1 (b), this method utilizes various BPNNs with different compression ratios which are used to compress/de-compress image blocks depending on the level of complexity in that block. In adaptive BPNN, at first, image blocks are divided into several classes according to their complexity and a network CODEC is chosen for each class with the compression ration related to the complexity of that class. All of the networks have identical structure, but they have different number of neurons in hidden layers, which will result in different compression ratios.

As it is shown in Figure-1 (b), to train the networks, the complexity level of each block is estimated by means of a value according to a complexity measure criterion. Then according to this complexity value, one of the available networks is selected. Each network is trained using its corresponding train data by Back-propagation algorithm.

After training and in evaluation phase, test image is fed into the network and compressed image is obtained in the outputs of hidden layer. These outputs must be quantized to the desired number of bits. If the same number of bits is used to represent input and hidden neurons, then the Compression Ratio (CR) will be the ratio of number of input to hidden neurons. In general, the compression ratio of the basic network illustrated in Figure-1 (a) for an image with n blocks is computed as equation (3).

$$CR = \frac{nNB_I}{nKB_H} = \frac{NB_I}{KB_H} \tag{3}$$

Where B_I and B_H are the number of bits needed to code the output of input and hidden layers, respectively. N and K are the number of neurons in the input and hidden layers, respectively. In adaptive approach, we assume to have M different networks with $k_1 \sim k_M$ neurons in hidden layer. In this case the compression ratio is as equation (4) that is obtained by modifying equation (3).

$$CR_a = \frac{nNB_I}{\sum_{i=1}^{n} B_H K_j^i + q} = \frac{NB_I}{(\frac{B_H}{n}\sum_{i=1}^{n} K_j^i) + \frac{q}{n}} ; 1 \le j \le M \tag{4}$$

In the above equation, K_j^i is the number of neurons in the hidden layer of selected network for the i^{th} block of image and $1 \le j \le M$. q is the number of bits that are needed to code the network number. In fact q is equal to the smallest positive integer such that $2^q \ge M$.

To identify for de-compressor which network is used in compressor stage, a code is assigned to each trained network and is transmitted or saved along the compressed image. The lower number of bits is preferred from the overhead view of point but on the other hand, the lower number of networks reduces the adaptively ability of the algorithm. In de-compressor, the compressed image is converted to a version similar to the original image by applying the hidden layer to output layer de-compression weights on outputs of hidden layer. The outputs of output neurons must be scaled back to the original grayscale range.

3. Complexity measurement methods

Complexity measure criterion is an important factor for adaptive approach. It affects the compression performance significantly. In the following three complexity measure methods are presented for calculating the detail level of image blocks. A criteria for complexity measurement should reveal the amount of information in an image block. Also, it should be able to discriminate the image blocks according to neural networks-based compression. In this paper, we have used three different criteria, Entropy, Pattern-based and Activity.

It is known that *Entropy* is a meaningful criterion to measure the amount of information in a set of symbols like an image. The entropy of an image block with N different gray-levels is calculated as (5). Where $P(x_i)$ is the probability of occurrence of gray-level x_i in this block.

$$Entropy = -\sum_{i=1}^{N} P(x_i) \log P(x_i) \tag{5}$$

An image block that has a higher *Entropy* value contains more information. It means that, to prevent more loss of data, that block should go through less compression. Also, lower *Entropy* value is related to higher compression ratio.

Activity is a subjective measure of activity in an image [3]. For an image block with N pixels (i.e. in size $\sqrt{N} \times \sqrt{N}$) *Activity* is defined as equation (6). In this measurement, low activity classes require networks with high compression rate and high activity classes need to maintain more data, which means that they should use networks with larger number of hidden neurons.

$$Activity = \sum_{i=2}^{\sqrt{N}} \sum_{j=2}^{\sqrt{N}} \left[\sum_{m=-1}^{1} \sum_{n=-1}^{1} (x_{i,j} - x_{i-m,j-n})^2 \right] \tag{6}$$
$$i, j = even; \ (m,n) \ne (0,0)$$

Although *Activity* is a good subjective criterion in complexity approximation and *Entropy* is a semantic measure for calculating the amount of information in a block of data, but in our usage of these methods for the learning purpose, we faced some difficulties. It is clear that the *Entropy* values of all image blocks in Figure-2-(5) to Figure-2-(8) are equal. This is true for the *Activity* values, too. So, all of those blocks will be compressed by the same network. Each of these blocks has different pattern and in order to obtain better compression ratio, it is better to assign these blocks to different networks.

To overcome this problem we have used another complexity measure named *pattern-based* method. In this method image blocks are classified into 16 patterns as shown in Figure-2, based on their patterns. This is done, by dividing a block into four equal sub-blocks. The division method is based on quad-tree representation of an image where cross-cut of blocks is not considered. Networks have various compression rates based on their related patterns. That is, patterns number 1 and 2 in the Figure-2 have maximum compression rates and patterns 3 and 4 have minimum compression rates.

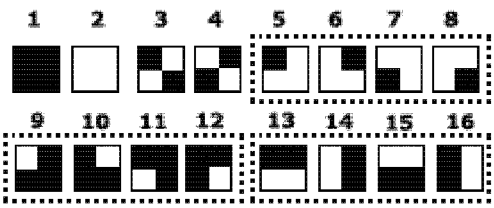


Figure-2 : Patterns used for classification of image blocks as a complexity measure criterion

4. Experimental Results

In this section we present the evaluation of compression ability of the basic network structure in Figure-1 (a) and adaptive approach of Figure-1 (b) with different complexity measure criteria. Also we have compared the adaptive method with JPEG standard coding algorithm. In addition to the Compression Ratio (CR), the performances of these methods are compared according to Peak Signal to Noise Ratio (PSNR) criterion. We have used 8 bits/pixel grayscale images in our experiments and selected image block size 8×8, which respectively results 64 neurons in input and output layers.

In evaluation of basic structure networks, fix compression ratio 4:1 (64:16) is used that correspond to 16 neurons in the hidden layer (i.e., 4×4 blocks) and 8 bits/neuron to code the outputs of hidden layer. For training the network, we have used the Lena image of size 256×256 and no overlapping between image blocks was used. Lena and three other images are used as test images. The results of CODEC with this structure are shown in Table 1. Clearly, this method has not provided acceptable quality for test images that were not in the training set. This fact is due to the data dependency problem in extracted transform resulted by trained weights.

Table 1: Compression results using basic EBPNN structure

Test image	PSNR (dB)
Lena	34.92
Camera man	26.67
Crowd	23.41
Pepper	22.20

Improved results are obtained using the adaptive approach. This method needs more training data due to higher number of network parameters. Training set selection is done in such a way that a wide range of complexities in images are covered and the sufficient numbers of training patterns are considered. For all three adaptive methods, the Lena image was presented in the training set and the other three test images were not. Like the basic structure, in training the adaptive networks, 8×8 image blocks are used and according to each complexity measure method, each block was given to its appropriate network. Blocks with low level of complexity were given to networks with higher compression ratio and vice versa.

Having the trained networks, given an input test image, the compression is done as the training routine. The images are divided into blocks and the compressor network is selected based on its complexity value. This method is called as *complexity-based* in experimental result of this section. Another choice in selecting the trained networks is to choose a network that minimizes an error measure criterion. We have named this approach *best-SNR* and used this network selection method by considering "maximization of signal to noise ratio" criteria for each block. Here, an image block is given to all networks and the network that results maximum SNR for the block is selected. Of course, this is a time consuming process; however it results the minimum error in reconstructed image.

Also, in order to reduce the chess-board effect in reconstructed image and improve its visibility quality, overlapping the adjacent image blocks is allowed. We have realized this idea by considering a %50 overlap between rows and columns of neighboring blocks. In the overlapped area, for each pixel, the average value of overlapped pixels is calculated.

4.1 Results using *Entropy* measure

In *Entropy* complexity measure criteria, six networks with 4, 9, 16, 25, 36 and 49 hidden neurons are used. This results 3 bits overhead for coding the number of networks. The results of this structure are shown in Table 2. The evaluation is done on 4 test images as shown in Figure-4 for Complexity-based and Best-SNR approaches with and without overlap for each image.

The ability of this method in good reconstruction of out-of-train images is considerable. The PSNR for all images in test set are close and due to the higher complexity of the Crowd image, its CR is less than others. Utilizing the block overlapping reduces chess-board effect and increases the PSNR about 1dB without affecting the CR. The Best-SNR network selection for test blocks results a considerable improvement in PSNR value and visibility quality as shown in parts (c) and (d) in Table 2. From the other point of view, the drawbacks of this approach are increasing CODEC time and decreasing CR. A remarkable note of results (a) to (d) in Table 2 is the distinction in PSNR improvement between Lena and Crowd images. This is because of difference in the complexity level of these images. Average complexity (*Entropy*) values for these two images are 3.71 and 5.32, respectively. Crowd is the most complex image in this test set. Using the Best-SNR has not provided significant improvement in the PSNR for this image. This is because the improvement resulted by the Best-SNR method is due to the property of this method in using networks with lower CR for less complex blocks in an image. The fact is that, in Crowd imag, 85% of the blocks have the complexity relate to network number 6 and only 5% of them relate to network number less than or equal to 4. So, less degradation in the CR of this image compare to other images can be also justified.

Table 2: Compression results by adaptive BPNN structure using *Entropy* complexity measure criteria

Network Selection	Test image	PSNR(dB)	CR_a
(a) *Complexity-based None-Overlapped*	Lena	35.50	2.76
	Camera man	35.11	2.83
	Crowd	37.92	1.71
	Pepper	35.62	2.20
(b) *Complexity-based Overlapped*	Lena	36.70	2.76
	Camera man	36.57	2.82
	Crowd	38.68	1.71
	Pepper	36.25	2.20
(c) *Best-SNR None-Overlapped*	Lena	43.10	1.79
	Camera man	39.24	2.04
	Crowd	38.11	1.70
	Pepper	38.06	1.78
(d) *Best-SNR Overlapped*	Lena	44.55	1.79
	Camera man	41.04	2.06
	Crowd	39.73	1.69
	Pepper	39.71	1.74

4.2 Results using *Activity* measure

In this method, 4 networks with 9, 16, 25 and 36 hidden neurons are used. Table 3 shows results of this adaptive method. The resulted PSNR in this method is better than *Entropy* for less complex images in (a) and (b) tests. These results indicate that this measurement method estimates the complexity better than *Entropy* only for less-complex images. The smaller number of networks in this method causes that the Best-SNR does not provide PSNR as good as *Entropy*-based method. On the other hand, keeping the data of the number of networks caused only 2 bits of overhead compared to 3 bits in *Entropy*-based method. Also regards to the selection of number of neurons in the hidden layer, this method provides better CR. The results of this method in case (d) indicates that Best-SNR, for complex images, Crowd and Pepper, do not provide high improvement. This is because the complexity of these images and same reasoning as previous section is correct.

Table 3: Compression results by adaptive BPNN structure using *Activity* complexity criteria

Network Selection	Test image	PSNR (dB)	CR_a
(a) *Complexity-based None-Overlapped*	Lena	42.04	11.94
	Camera man	37.79	9.25
	Crowd	36.15	4.78
	Pepper	34.28	6.93
(b) *Complexity-based Overlapped*	Lena	43.12	11.79
	Camera man	38.24	9.16
	Crowd	36.41	4.75
	Pepper	34.93	6.67
(c) *Best-SNR None-Overlapped*	Lena	43.99	7.96
	Camera man	38.56	7.47
	Crowd	36.58	4.46
	Pepper	35.43	4.82
(d) *Best-SNR Overlapped*	Lena	45.09	7.93
	Camera man	39.04	7.38
	Crowd	36.82	4.38
	Pepper	35.74	4.81

Table 4: Compression results by adaptive BPNN structure using *Pattern-based* complexity criteria

Network Selection	Test image	PSNR (dB)	CR_a
(a) *Complexity-based None-Overlapped*	Lena	39.11	6.67
	Camera man	34.44	6.69
	Crowd	30.66	5.11
	Pepper	32.01	6.11
(b) *Complexity-based Overlapped*	Lena	40.20	6.67
	Camera man	34.80	6.73
	Crowd	32.00	5.10
	Pepper	32.86	6.19
(c) *Best-SNR None-Overlapped*	Lena	44.00	3.24
	Camera man	38.60	3.70
	Crowd	36.91	2.90
	Pepper	36.08	3.00
(d) *Best-SNR Overlapped*	Lena	46.76	3.26
	Camera man	40.45	3.71
	Crowd	39.41	2.89
	Pepper	38.14	3.01

4.3 Results using *Pattern-based* measure

The simulation of this method is done using 16 networks, one network for each pattern in Figure-2. The CRs related to these patterns are 16:1 for patterns number 1 and 2, 4:1 for patterns numbers 6~8 and 9~12, 7.1:1 for patterns number 13~16 and 2.5:1 for patterns 3 and 4. The CODEC results of this method are shown in Table 4. These results are almost in the same direction of two former methods. The overlapping increases the PSNR about 1dB and Best-SNR network selection method has resulted higher quality reconstructed images even for complex images. This is because of the higher number of networks used in this method compared to previous methods. Also due to the network structures the CR of images is higher than Entropy-base structure and is lower than Activity-based. There are 4 overhead bits in this method to code the network number. In addition, the reduction of CR value with Best-SNR is reasonable because of the higher number of networks.

4.4 Comparison of complexity measures

The compression of three proposed adaptive methods using their results is not so reasonable because each one uses different number of networks and different structures in its networks. As the results of tables 3, 4 and 5 show, the *Entropy* presents good estimation of complexity for complex images. On the other hand, *Activity* performs better estimation for less-complex and simpler images. This method has resulted higher CRs due to its network structure.

The *Pattern-based* complexity measure method has resulted almost the same PSNR with higher CR than *Entropy* and lower CR than *Activity*. For better comparison of these methods and considering both CR and PSNR together, we have compared these three methods in their best performance with JPEG standard. The results are shown in Figure-3. It illustrates the ability of adaptive compression methods compared to the standard JPEG algorithm. This comparison is done in the same bit rate for each method. These results show the achievements and even superiority of ANN-based compression to this compression standard.

In addition to the compression results, the three proposed adaptive methods are different from fundamental principle point of view. *Entropy* is a measure of uncertainty of a random variable. The *Activity* is an intuitional method for estimating the complexity of an image block using the difference between each pixel and its neighbors. These two methods do not discriminate the place or direction of the complexity and give an average value of complexity.

On the other hand, *Pattern-based* method classifies the image block into some predefined patterns. Actually, this method is not a complexity measure criterion and does not have the mentioned problem of other methods, but it indicates another problem. It does not consider grayscale values exactly and finally maps each sub-blocks into one block or white pattern.

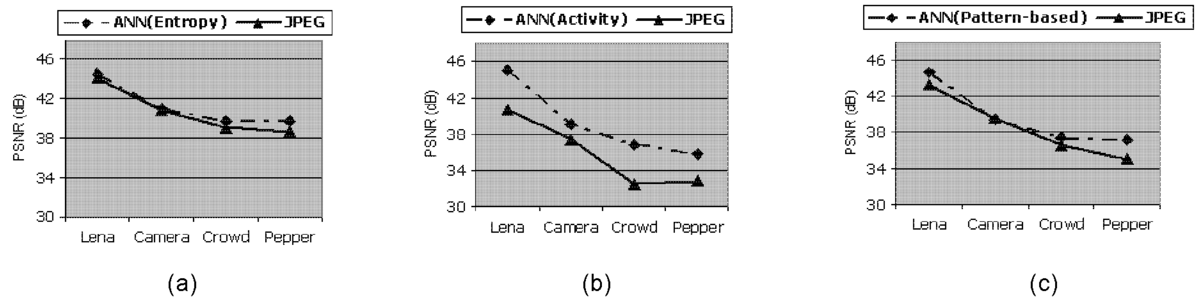


Figure 3: Comparison of the adaptive methods (a) Entropy (b) Activity and (c) Pattern-based with the JPEG compression standard

5. Summary and Conclusion

We have reviewed the use of multi-layer preceptoron Neural Networks for image compression. Since acceptable result is not obtained by compression with one network, an adaptive approach is used such that different networks for different image blocks regarding to their complexity values. Three complexity measurement methods *Entropy*, *Activity* and *Pattern-based* are presented and evaluated. Our experimental results showed that better visual quality is obtained by overlapping neighboring image blocks. Also selecting images with Best-SNR criteria rather than the complexity criteria provides higher image quality and better PSNR. Higher number of networks provides better performance in Best-SNR approach but this will result in lower CR. Comparing results with standard JPEG algorithm showed better performance for our method both with PSNR measure and visibility quality.

Selection of the number of hidden layer neurons in this research is done based on the visual and intuitive sense of complexity of the related patterns and they can be chosen in other manners. The mechanism of network selection is not optimized here and it is expected that using larger number of networks and selecting optimum compression ratio for networks, could provide better results.

Among three proposed methods the *Entropy* and *Activity* do not use the orientation of patterns and the *Pattern-based* does not use the gray-levels values. These problems can be solved by utilizing the gray-level values in the *Pattern-based* method which can be realized by combining the *Activity* or *Entropy* with *Pattern-based* method.

In addition, from a compression method viewpoint, the rate-control ability or having rate-distortion function is an important factor. However, it seems that ANN-based methods are not flexible in controlling the compression ratio, but it is possible to have a set of trained networks with different compression ratios rather than one network in each case.

References

[1] N.Sonehara, M.Kawato, S.Miyake, K.Nakane, Image compression using a neural network model, International Joint Conference on Neural Networks, Washington DC, 1989.

[2] S. Carrato, G. Ramponi, Improved structures based on neural networks for image compression, IEEE Workshop on Neural Networks for Signal Processing, New Jersey, September 1991.

[3] S. Carrato, S. Marsi, Compression of subband-filterd images via neural networks, IEEE Workshop on Neural Networks for Signal Processing. August 1992.

[4] G. Qiu, M. Varley, T. Terrel, Image compression by edge pattern learning using multilayer perceptron, Electronic letters, Vol 29, No 7, April 1993.

[5] J. Jiang, Image compression with neural networks -A survey, Image Communication, ELSEVIER, Vol. 14, No. 9, 1999.

[6] C. Cramer, Neural networks for image and video compression: A review, European Journal of Operational Research, Vol. 108, July 1998.

[7] A.Namphol, S.Chin, M. Arozullah, Image compression with a hierarchical neural network, IEEE Trans. Aerospace Electronic Systems Vol. 32 No.1, January 1996.

[8] J. S. Lin, S.H. Liu, A competitive continuous Hopfield neural network for vector quantization in image compression, Engineering Applications of Artificial Intelligence, Vol. 12, 1999.

[9] G. Pavlidis, A. Tsompanopoulos, A. Atsalakis, N. Papamarkos, C. Chamzas, A Vector Quantization – Entropy Coder Image Compression System, IX Spanish Symposium on Pattern Recognition and Image Processing, 2001.

[10] C. Amerijckx, J. D. Legaty, M. Verleysenz, Image Compression Using Self-Organizing Maps, Systems Analysis Modeling Simulation Vol. 43, No. 11, November 2003.

[11] S. Costa, S. Fiori, Image compression using principal component neural networks, Image and vision computing, Vol. 19, 2001.

[12] M. Egmont-Petersen, D. de Ridder, and H. Handels, Image processing with neural networks - a review, Pattern Recognition, vol. 35, pp. 2279-2301, 2002.

[13] A. Rahman and Chowdhury Mofizur Rahman, "A New Approach for Compressing Color Images using Neural Network", Proceedings of International Conference on Computational Intelligence for Modeling, Control and Automation - CIMCA 2003 , Vienna, Austria, 2003.

[14] Sonja Grgic, Marta Mrak, Mislav Grgic, Comparison of JPEG Image Coders Proceedings of the 3rd International Symposium on Video Processing and Multimedia Communications, VIPromCom, pp. 79-85, Zadar, June 2001.