Image Compression Technique: A Neural Network Approach

Engr. Orland Delfino Tubola
Department of Computer Engineering
Polytechnic University of the Philippines
Sta. Mesa, Manila
tubology@gmail.com

Gladys Tapar Obmerga
Department of Computer Engineering
Polytechnic University of the Philippines
Sta. Mesa, Manila
gladysobmerga19@gmail.com

Harold Taroy Asuncion
Department of Computer Engineering
Polytechnic University of the Philippines
Sta. Mesa, Manila
asuncionharold@gmail.com

Pinky Solas Pal-lingayan
Department of Computer Engineering
Polytechnic University of the Philippines
Sta. Mesa, Manila
pinkypallingayan@gmail.com

Abstract - **Image Compression based on artificial neural network is presented in this paper. Researchers aimed to develop it using MATLAB by presenting it in a graphical user interface window wherein the pixels of the image is used as input values and the target values was given so that the desired mean square error will be obtained. And then the hidden layer output attained from the network will be the compressed image. It breaks down large images into smaller windows and eliminates redundant information such as coding redundancy, inter pixel redundancy and psycho visual redundancy. Back propagation method was used to train the network and for the minimization of the error. Performance of the network has been tested by the use of some standard grayscale test image like Lena, cameraman and pepper with the dimensions of 512x512, 256x256 and 128x128 and a bit depth of 8. Other properties aside from the given value of the dimensions and bit depth will not be considered in the study. It is shown that the developed program for the said network and the training algorithm used which is the back propagation algorithm, provides a high compression ratio for the image with minimal distortion.**

Keywords – Neural Network, Image Compression, Back Propagation Network

I. Introduction

It is said that picture is worth a thousand words. This clarifies the difference between the human's capability to perceive linguistic and visual information. Likewise, visual representation tends to be more efficient

than the spoken or written word when it comes to perceiving information. Language, in an evolutionary perspective, develops instantaneously and is expressed in only a single species. Unlike in visual representation, in one form or another, has been existing since hundreds of millions of years ago and is shared by a numerous organisms. But why should there be such a difference between the two representations in such an advances being as ourselves?

Nowadays, photos are being handled by the latest gadgets that has been developed over the years such as phones, laptops, desktops and even in company computers. These hi-tech gadgets, especially laptops, gain the capacity to run any programs like software that process digital images. And these software rapidly advances in its smartest and simplest way. In relation to these development, today's digital imaging software have reach the level of photo editing and even its 3-D representation. Its user interfaces intended to be friendly to advance users as well as first-time users.

The development of internet raise the growth of digital imaging. Online editing, sharing and viewing of photos are now allowed on the internet. Thus, online websites like Instagram, Facebook, Pinterest are emerging one after the other to give humans the capability to share their photos and to viewed other's phots also whether they are professionals or not. This have changed the way people's understanding in the field of photography.

Photography has gone from being a luxury medium of communication and sharing to more of a fleeting moment in time. Subjects have also changed. Pictures used to be primarily taken of people and family. Now, we take them of anything. We can document our day and share it with everyone with the touch of our fingers. *(Wikipedia, Digital Imaging)*

As smart devices continue to transpire, digitals images, when it comes to the transmission and sharing, have increased excessively. The more these devices includes cameras and provides the users to share the captured images directly to the Internet, the more these storage devices are grasping the necessity of effectual storing of huge amount of image data. These contributes to inadequate bandwidth of network and storage of memory device. Therefore, the theory of compressing the data becomes more and more significant for reducing the data redundancy save more hardware space and transmission bandwidth.

Methods such as image compressions has been an active area of research since the beginning of digital image processing. The idea of this is that it converts the original image with few bits. Since images can be represented in a two-dimensional signal, a lot of digital compression techniques are being studied in order to meet the needs of image compression. Without significantly reducing its quality, it is useful because of the fact that it reduces the bandwidth consumed during the transferring of images from one medium to another and the cost it takes in a hard disk's space. It is beneficial in many applications especially in progressive transmission, image browsing and multimedia application. Its general objective is to obtain the image quality and yet occupy less space

This study focuses on compressing images using neural network. This chapter discuss about the main objectives of the study and the limitations that it can handle. And also the problem that is being experienced today that serve as the basis for the researchers to conduct this study. Chapter 2 briefly summarizes some of the works that have been done by other researchers which is correlated in this study. Chapter 3 provides the ideas and concepts for image compression which helps the researchers to gain understanding on their chosen topic. Chapter 4 is all about the approaches made to implement the concepts gathered in the study. It introduces some diagrams which can help to have a deeper understanding about the basis for the procedures used. Chapter 5 summarizes the results and discuss the core ideas developed and analyzed by the researchers. Chapter 6 stated some of the constraints experienced by them during the process of developing the system and some standards to be considered in conducting the study

II. Methodology

The chapter will discuss about the procedures and techniques used by the researchers.

A. Interfacing the Image Compression System

a. Creating a Graphical User Interface Window (with GUIDE)

The researchers aim to create their system interface using MATLAB Graphical User Interface Development Environment. To create a GUI, click on New on the Home tab, then select the Graphical User Interface. The GUIDE Quick Start window will pop up on the screen. Go to the 'Create New GUI' tab then select 'Blank GUI (Default)'. Save the new blank GUI file, with an extension '*.fig' by checking the 'Save new figure as' below the tab and browse or create the folder wherein the GUI will be saved. After this, an editor and a '*.fig' window will be shown on the screen. This is where the desired GUI window can be edited: the '*.fig' window for the layout of the GUI and the editor for the function and process of the GUI.

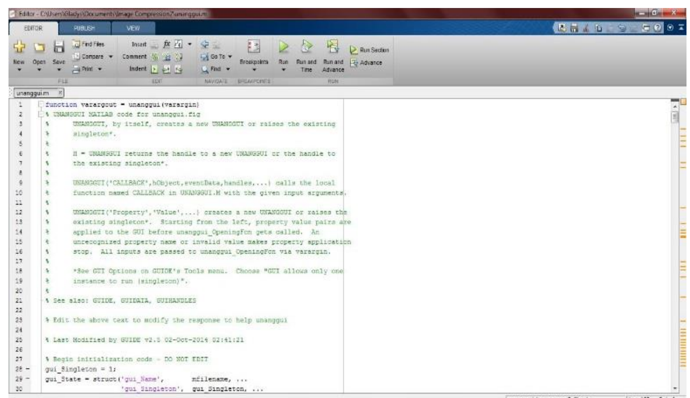


Figure 1. The Editor Window

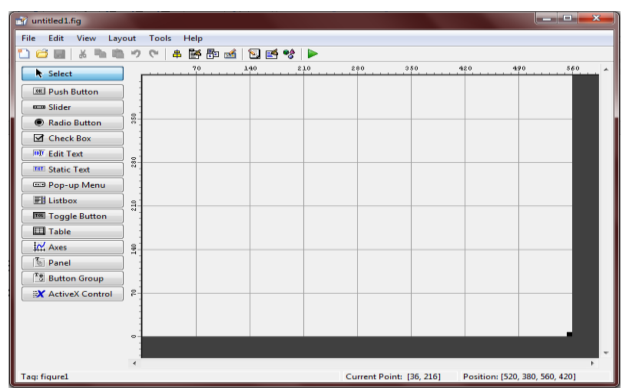


Figure 2. Graphical User Interface Development Environment (GUIDE)

Select buttons, axes (for viewing image or for plotting function), or static texts, and modify the string and tag of each selected object on the property inspector to call its functions on the editor. The property inspector allows to modify the necessary adjustment to be made on the object including the size of the object, its color, button function and etc.

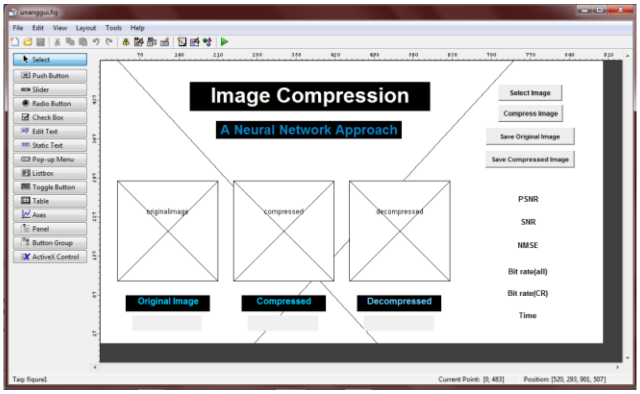


Figure 3. Image Compression GUI layout in MATLAB GUIDE

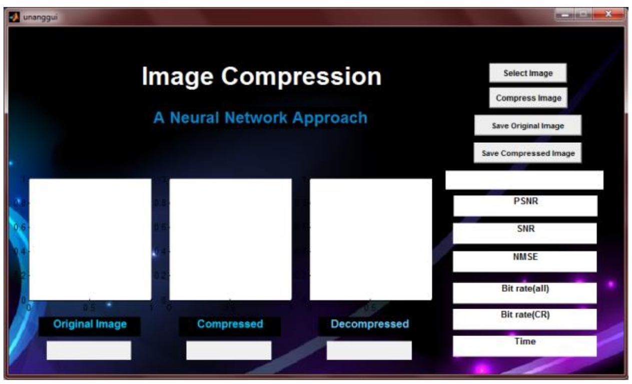


Figure 4. The researcher's Image compression system GUI

B. The Image Compression Process

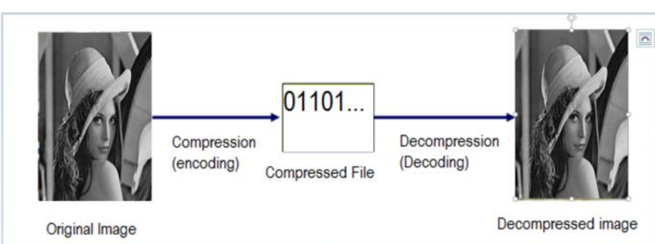


Figure 5. Block diagram of Image Compression

a. Constructing a Network Architecture

Network architecture has been defined by the researchers to establish the system.

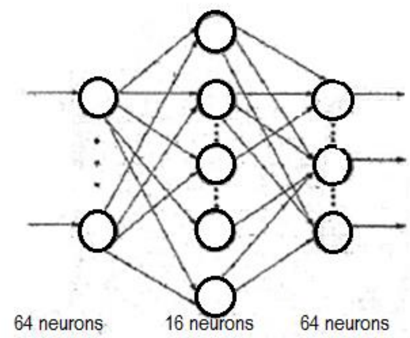


Figure 6. Typical Neural Network Architecture (Ilic and Berkovic)

The chosen number of input and output neurons should be the same and decision about its quantity must be assessed properly because it is solely related to the whole system that is being created. In image compression, for example, selected number of input and output neurons depends on the number of two dimensional array of pixels of an image which each elements of pixels are values from 0 to 255. Then the constructed network will compress the grayscale image. Blocks of size 8x8 pixels are extracted from the image, which means that values of 64 pixels used in network training are presented to the network input layer. Therefore, input layer, same with the output layer, has 64 neurons and choice about the number of neurons in the hidden layer is free but it has to be less than the number of neurons in input layer.

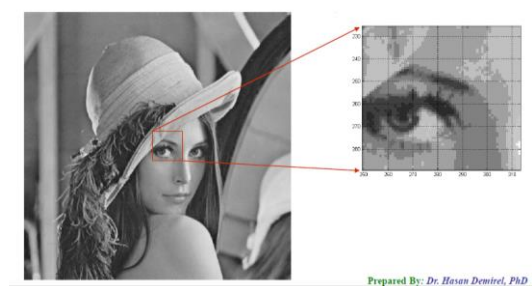


Figure 7. Lena's extracted Block of Image

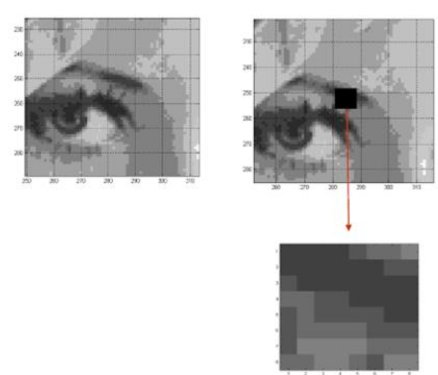


Figure 8. 8x8 Blocked to be processed

b. Creating Input Values

Image is divided into 8x8 pixels blocks and this will be the input for the network. The activation function, which is the sigmoid function requires each neuron to be in the range of 0 to 1, because normally, the image have input values ranging from 0- 255. The process of linearly transformation of image values from 0-255 into another range appropriate for the network is called Normalization. It is attained through the division of pixel values with its biggest possible value (255). Network is more operative when the input and output values are limited only to the interval [0,1].

Initially, image is a 2-dimensional blocks. Therefore linearization is required to transform the 2-dimensional block image into 1-dimensional vector for the representation of the input into the neural network. Linearization involves the methods of scanning rows, columns or labyrinth. To simplify the learning process, segmentation is used. It is the process of dividing it into non overlapping blocks with equal size.

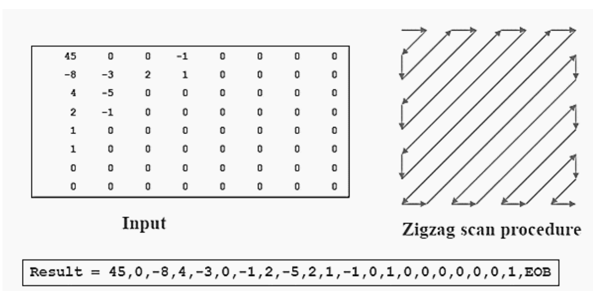


Figure 9. Process of creating input values

c. Training Pairs Preparation

For the training of the network to be possible, two weight matrices must be represented. The first matrix, which the researchers assigned in a variable v, represents the weight connections between the input and hidden layer. And the second matrix w, is for the connections between the hidden and the output layer. This has been initialized into small random values because these training pairs affect the whole process of training the network. Its value, as other researchers have utilized also, must be ranging between numbers having a values of 0.1 to 0.9, but it still depends on the user of the network. Large values can lead to the instability of the network while smaller values can cause excessively slow learning

d. Network Training

The researchers chose the appropriate learning algorithm for the network which is the back propagation algorithm. Algorithm is responsible for the training of the network by modifying its weight. The desired output has been provided so that the weights circulating in the network will be updated by the algorithm according to the quadratic error between output and the desired one. The error must be in its highest minimum value in order for the network to achieve its goal. The weights matrices are being remembered during the training process. These stored parameters provides the possibility of presenting to the network a completely new image that is also still need to be processed and in order to apply algorithm of compression and decompression.

The algorithm used gradient of preformation function in order to identify the way how to modify weights and to minimize the function. For the weights w in the network according to the gradient rule, new value is calculated following the expression:

$$W_t\ (new) = W_i\ (old) + \alpha(Y - O)\ X_i$$

Where:

a - constant that maintain the learning rate

X_i - input of the neuron i

Y - calculated output from the neuron

O - expected output from the neuron

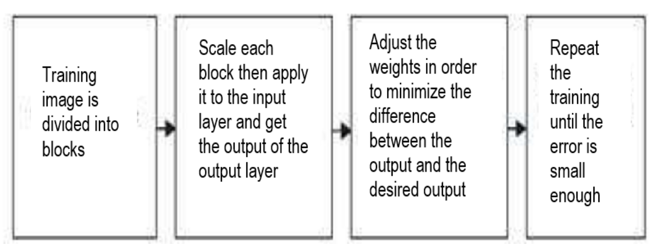


Figure 10. Block Diagram of Network Training

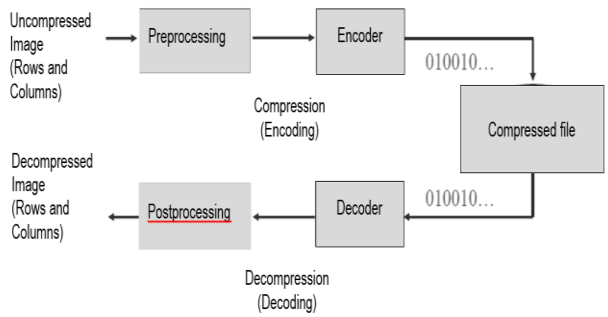


Figure 11. Image Compression Model

e. Compression

To remove the redundancies and to create the main process of compressing the image, the encoder is designed. After the network training, new inputs and expected outputs (image that is going to be compressed) are transferred into the network input layer. The first stage of encoding is called mapping. A mapper is used to reduce some data redundancies like spatial and temporal redundancies. This operation is generally reversible and may or may not reduce directly the amount of data required to represent the image. Some methods used for mapping are run length coding, predictive coding and transform coding. With the used of the stored weights, hidden layer output is calculated and then attained values are quantified with 8 bits and remembered.

Next stage for compressing the image is called the quantizer. Quantizer reduce psychovisual redundancies. It is irreversible unlike the mapper and the symbol encoder. Keeping the irrelevant information is its goal out of the compressed representation. It must be omitted when the desired error-free compression has been attained. Uniform quantization and nonuniform quantization are the examples for quantizer.

After the data is being reduce by the mapper and the quantizer, it will now reduce the coding redundancies stored in the image using a symbol encoder like Huffman coding, arithmetic coding and LZW. This is the third and the final stage wherein it generates a fixed length code to represent the quantizer output and maps the output in accordance with the code. And same with quantizer, this operation is irreversible. The figure 10 below shows the block diagram of an encoder

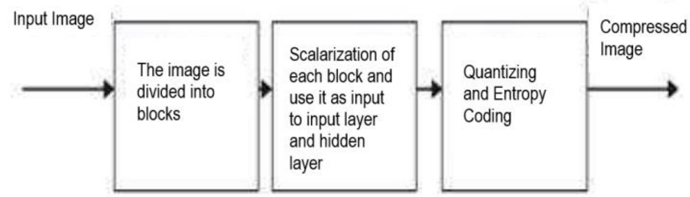


Figure 12. Encoding Steps (Mehare, Shibu 2012)

f. Decompression

Decompression simply reverse the path of compression. The reconstruction of the compressed image involves the reading the stored data (compressed image) to set as hidden layer outputs, and then, again using the previously stored weight matrices, calculate the network output with the input to the output layer and weights between hidden and output layer. This process of reconstruction the image is called post processing. The decoder (decompression) has two stages: a symbol decoder and an inverse mapper. They only perform the operation of encoder's symbol encoder and mapper in reversible way. The reason why decoder do not have quantizer is because the quantizer is irreversible, so this block is no longer included in the general decoder model.

To decompress the image; first decode the entropy coding then apply it to the output of the hidden layer and get the output of the OL scale it and reconstruct the image. Fig. 11 show the decoder block diagram.

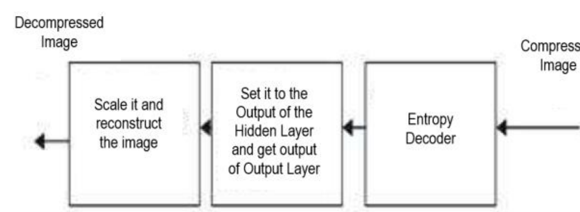


Figure 13. Decoding steps (Mehare, Shibu 2012)

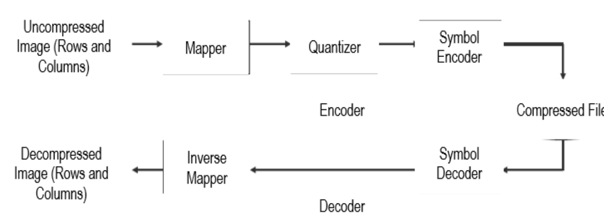


Figure 14. Process of encoding and decoding

III. Experimental Results

After the system has been modified and tested, this part of the study will be discussing about the topics connected and the result of the process made by the researchers based on some given criteria and also the final insights of the researchers after all these processes

Researchers made use of the Lena, cameraman and pepper as test images. The results obtained using the network with 64 neurons in input and output layer and 16 neurons in its hidden layer will be shown in the succeeding tables by means of fidelity criteria. Fidelity criteria is used to measure information loss in image compression and it is basically divided into two types: subjective fidelity criteria and objective fidelity criteria.

A. Subjective Fidelity Criteria

It is measured by human observation. The table below shows the subjective differences of the original image, the result of the encoding process which is the compressed image and the decompressed image.

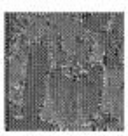
Image name	Original Image	Compressed Image	Decompressed Image
Lena			
Cameraman			
Pepper			

Table 1. Compression and Decompression results of sample images

This result indicates that decompression of images is possible but the quality, based on this type of criteria, will depend on the dogmatic observation of human. Researchers have provided a subjective evaluations as the basis for rating the image quality.

Impairment	Quality	Comparison
5 - Imperceptible	A - Excellent	+2 much better
4 - Perceptible, not annoying	B - Good	+1 better
3 - Somewhat annoying	C - Fair	0 the same
2 - Severely annoying	D - Poor	-1 worse
1 - unusable	E - Bad	-2 much worse

Table 2. Sample Subjective Evaluation

Impairment test is used to evaluate how bad are the decompressed images while quality test is for how good the decompressed images are. Comparison test weigh the result between the original image and the decompressed image

B. Objective Fidelity Criteria.

It measures mathematically the amount of error between the original image and the compressed/decompressed image. The researchers made use of some formulas in measuring the quality of the decompressed images. Such formulas used were signal-to-noise ratio (SNR), peak signal-to-noise ratio (PSNR) and normalized-mean-square-error (NMSE). Table 2 summarizes the comparison of the properties of the images after the process of compressing and decompressing the image, which have utilized by the researchers to calculate some quality measures to be discussed further in this section.

before compression					after compression		
filename	dimension	bit depth	size	type	size	saveas	Time
Lena	512x512	8	256 kb	bmp	24.3kb	jpeg	14 seconds
pepper	256x256	8	39.2kb	png	8.69kb	jpeg	29 seconds
cameraman	256x256	8	65kb	bmp	6.39kb	jpeg	10 seconds

Table 3. Detailed properties of the images before and after compression

It is shown that the file size of the original images is reduced in a certain ratio. Image of Lena, from the original file size of 256kb, became 24.3 kb after the compression process. Same with the other test images like cameraman and pepper in which have reduced from 39.2kb to 8.69kb and 65kb to 6.39kb.

a. Mean Square Error

The mean squared error is computed based on the target image and the decompressed image. It should be as small as possible so that the quality of the reconstructed image is near to the target image. For ideal decompression, the value of MSE should be zero. The formula used to compute the MSE is

$$\text{MSE} = \frac{1}{NM}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[\hat{f}(x,y) - f(x,y)\right]$$

The table below shows the result of the MSE of each images.

Image	MSE
Lena	0.010125
cameraman	0.008943
pepper	0.047913

Table 4. Calculated MSE of the test images

Based on the given results, having a values between 0.01 to 0.04, it is shown that the reconstructed images are near to the target images since the value of the MSE of each test images is close to zero .

b. Peak signal-to-noise ratio

The PSNR value is the difference between the original image and its decoded image. The larger the PSNR value, the better will be the decoded image quality. The PSNR computes by the following equation:

$$\text{PSNR} = 10\log_{10}\frac{255^2}{MSE}$$

The table below shows the result of the PSNR of each images

Image	PSNR
Lena	29.3509db
cameraman	26.2659 dB
pepper	18.8124 dB

Table 5. PSNR of each sample test images

If the results obtained a lower MSE and a high PSNR, it literally means that the decoded image is a better one.

c. Compression ratio

Compression ratio is calculated as the ratio between the original uncompressed image file and the decompressed image. This is basically the data fed to the input layer neurons and the out from the hidden layer neurons. The researcher's network has 64 input neuron and 16 hidden layer neuron, thus using the formula

$$\text{CR} = \left(1 - \frac{H}{I}\right) * 100\%$$

Where

H – Number of hidden layer neurons (16)

I – Number of input layer neurons (64)

The compression ratio of the network is 75%.

Another way of solving the compression ratio is based on the file sizes of the original and compressed image. The table below shows the calculated ratio based on the formula

$$\text{CR} = \frac{\text{uncompressed file size}}{\text{compressed file size}}$$

Image	original file size	compressed file size	compression ratio
lena	256kb	24.3kb	10.535
cameraman	39.2kb	8.69	4.5109
pepper	65kb	6.39kb	10.1721

Table 6. Compression Ratio between the original file size and the compressed file size

d. Bit Rate

This can be calculated as the average number of bits per pixel for the encoded image. The table provided is the result of the bit rate in the network. Both the bit rate and the distortion measure must be showed for any meaningful comparison of the network's performance.

Image	Bit Rate
lena	2.0313 bpp
cameraman	2.125 bpp
pepper	2.125 bpp

Table 7. Bit rate of the test images

IV. Conclusion

Neural network based image compression technique has been used for the successful compression and decompression of the image. Based from this study, it can be concluded that image compression system can be applied using artificial neural network. Neural networks are well suited to the task of processing the image data because of its organizational structure and it provides an internal representation of data, which is needed in the process of image compression. MATLAB is used for the system, wherein it is capable of handling matrices of images. Neural Network approaches to image compression have been shown to perform as well as or better than standard approaches thus it should be considered as an alternative method to other traditional techniques of image compression.

V. Bibliography

Anjana B. and Mrs Shreeja R.: ‘Image Compression: An Artificial Neural Network Approach’

K. M. Ashikur Rahman and Chowdhury Mofizur Rahman: ‘A New Approach for Compressing Color Images using Neural Network’ Department of Computer Science and Engineering. Bangladesh University of Engineering and Technology, Dhaka-1000, Bangladesh

J. Jiang: ‘Image compression with neural networks: A survey’

S .S. Panda, M.S.R.S Prasad, MNM Prasad, Ch. SKVR Naidu: ‘Image Compression using Back Propagation Neural Network’

Dubravka Ilic, Ivana Berkovic: ‘Grayscale Image Compression Using Back Propagation Algorithm’

K.P.Paradeshi : IMAGE COMPRESSION BY EZW COMBINING HUFFMAN AND ARITHMETIC ENCODER

Jamshid Nazari and Okan K. Ersoy: Implementation of back-propagation neural networks with MatLab

Vaishali Mehare, Sini Shibu: A Neural Network Approach to Improve the Lossless Image Compression ratio

Vijayvargiya, Silakari and Pandey : A Survey: Various Techniques of Image Compression

Mahmud Hasan, Kamruddin Md. Nur: A Lossless Image Compression Technique using Location Based Approach

Wilford Gillespie : Still Image Compression Using Neural Network

G.M.Padmaja, 2 P.Nirupama : Analysis of Various Image Compression Techniques

Juan A. Ramírez-Quintana, Mario I. Chacon-Murguia and Jose F. Chacon-Hinojos: Artificial Neural Image Processing Applications: A Survey

M. Venkata Subbarao, N.Sayedu Khasim, Jagadeesh Thati and M. H. H.Sastry : Hybrid Image Compression using DWT and Neural Networks

Robert D. Dony : Neural Network Approaches to Image Compression

http://en.wikipedia.org/wiki/Digital_imaging