A Novel Technique for Image Compression in Hand Written Recognition using Back Propagation in Neural Network

P.Padmasree
M.Tech student ,SCSE
VIT Chennai
India

Maheswari.R
Assistant Professor(Sr),SCSE
VIT Chennai
India
maheswari.r@vit.ac.in

Abstract-The handwritten symbol recognition plays an important role in present communication systems. In the data communication systems, all the data have to be recorded, encoded and will be communicated with other systems. Presently this communication system extracts the digital and non-digital information from printers, scanners, and touch screens and from the image processing techniques which will be fed into the different pattern recognition algorithms. The digital data communication is easier rather than non-digital data communication like hand written symbols due to the variation of styles, size and shapes of the handwritten symbols from person to person so we have to compress the image which helps to reduce the storage space and the transmission cost . This paper presents implementation of handwritten symbol recognition using back-propagation algorithm in neural networks using MATLAB with training dataset which will give the compressed image exactly with improved resolution. This algorithm helps to increase the performance of the system.

Keywords- Handwritten symbol recognition, Neural Networks, Back-propagation.

I.INTRODUCTION

Handwritten symbol recognition plays an important role in data communication systems. The major problem in handwritten symbol recognition is the symbols (characters, digits) may be in different styles and sizes depending on the users. Sometimes it may exceeds the existing pixel range of the system so the resolution will be less to recognize the handwritten symbols. To overcome this issue each character (or) symbol are being considered as image and compress that image with the help of neural networks.

The compression of the digital image is become a crucial problem today. Data Communication wouldn't be possible without high compression rate. Numerous researches are going on image compression; numerous image compression techniques were introduced. In those artificial neural networks plays a major role.

II.THE IDEA OF NEURAL NETWORKS FOR IMAGE COMPRESSION

The basic idea of using Neural networks for image compression is to build a network such that the input and the output layers should contain the same number of neurons and the these neurons will be connected to the hidden layers with smaller number of neurons suppose the Neural networks with 64 input neurons, 4, 8, (or) 16 neurons in the middle layer and 64 output neurons.

There is a relation existing between the input layer and hidden layer which is defined as compression rate. The output should contain the neurons equal to its input neurons. The goal of such compression is to reconstruct its input. The network training starts with the initialization of weights to the input layer. The weights in the network are modified iteratively at the training time until to satisfied level of deviation between calculated and expected output is achieved. Any way there will be a compression loss in the existing techniques. The original image cannot be reconstructed completely and the error cannot be minimized. To overcome this issue, this paper presents a back propagation algorithm to minimize the error i.e. from the desired output the error is identified and giving it back to back i.e.to the hidden layers and changing the weights until the desired output is acheived.

III NEURAL NETWORK

The network mentioned in this paper is implemented in software package Mat lab

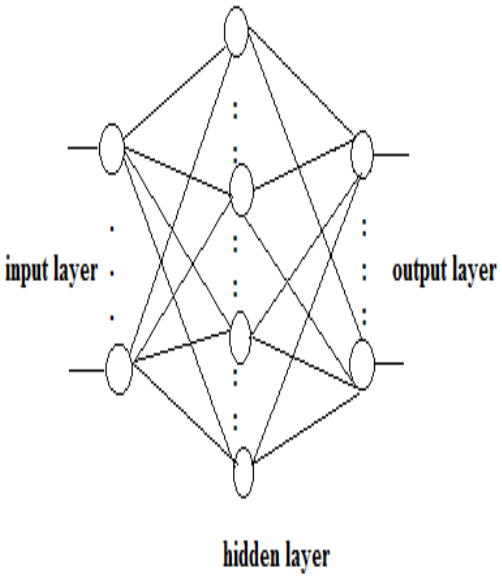

Figure 1 : Typical neural network architecture

The input layer is constructed in such a way that suppose if it has 64 neurons in the input layer the same number of neurons will be in the output layer. The decision about the number of neurons in the hidden layer will be less than the number of neurons in the input layer (4,8,16)

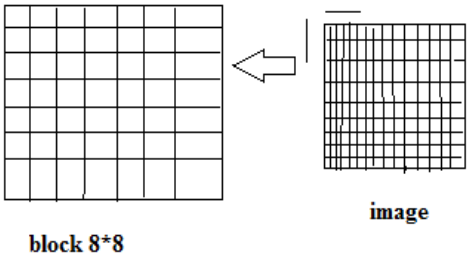

Figure 2 : Extracted blocks of image

IV IMPLEMENTATION OF BACKPROPAGATION ALGORITHM

The back propagation algorithm consists of various steps.

i) Each input is multiplied by the corresponding weights and sum up all these inputs with their weights. It will compute the total input X_j by the formulae

$$Xj=\sum_i (YiWij) \quad (4.1)$$

Y is the active level of the previous level and $W_{ij.}$ is the weight connected between the i^{th} and $j^{th\ unit}$. And this X_j will be passed through the hidden layer and it will be given to the sigmoid function and it will be scale the output between 0 and 1. The sigmoid function is given equation 4.2

$$Y_j=1/(1+e^{-xj}) \quad (4.2)$$

When the output is generated it will be compared with the original output and it will calculate the error between original output and the desired output .Once the activities of all output levels determined and then it will compute the final error (E).

$$E=1/2 \sum_j (Y_j-d_j)^2 \quad (4.3)$$

Y_j is the original output and d_j is the desired output once the error is computed it has to minimize this error by giving the error back to back and changing the weights until the desired output is achieved. This process is incorporated with three steps like

Step 1: Calculate how fastly the error changes as the activity of the output unit is changed then calculated the error derivative it is the difference between the actual and desired output.

$$EA_{jj}=Y_j-d_j \qquad (4.4)$$

Step 2: After that compute how fastly the error changes as the total input received by an output unit gets changed. It will be obtained from the step 1 by multiplying it with a rate at which the output is going to change with its total input is changed

$$EI_j=EA_jY_i(1-Y_i) \qquad (4.5)$$

Step 3: Compute how fastly the error is going to change as a weight on the connection into an output unit which gets changed and this will be getting from the step 2 multiplying it with the rate at which the input is going to change with its total weight changed

$$EW_{ij}=EI_jY_i \qquad (4.6)$$

Compute how fast the error changes as the activity of the previous layer is going to be changed there will be an effect on the output. in this way we will do this computations back to back until the error will be minimum.

V. PROPOSED TRAINING ALGORITHM USED IN THE BACK PROPAGATION ALGORITHM

The steps are as follows

1. Initialize the weights randomly
2. Find the actual output from those input neurons and weights this is the forward phase
3. After obtaining the output calculate the difference between the actual and desired output this is nothing but called as the error.
4. According to that error adjust the weights in such a way that to reduce the error .This is called as the backward phase
5. Repeat these steps for all training vectors
6. Repeat these steps until the error is small

A. Forward Propagation

The activations propagate from the input layer to the output layer and we can get the output.

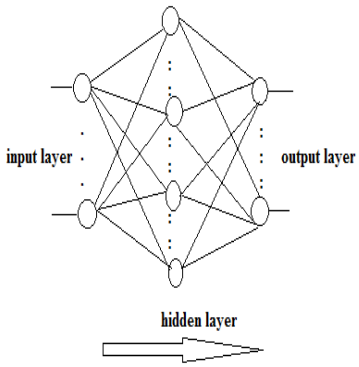

Figure 3: forward propagation

B. Backward Propagation

The error between the original and desired output will be calculated and this will be propagated backward so it will modify the weights and bias values until we will get the acceptably small error

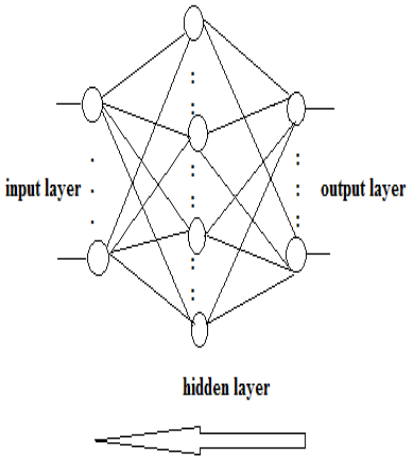

Figure 4: back ward propagation

VI.RESULT AND DISCUSSION

The compressed image quality will be measured with some parameters in those mainly used parameters are Root mean square error (RMSE) and peak signal to noise ratio error (PSNR), compression rate (CR) .The peak signal to noise ratio value is used to calculated the difference between the decoded image and the original image .If the PSNR ratio is high there will be better quality in the decoded image.

$RSME= [1/MN\sum_{i}^{M-1}\sum_{j}^{N-1}[f^{\wedge}(i,j)-f(i,j)]^2]^2$

(6.1)

$PSNR=10 \log_{10}[MN/RMSE^2]$ (6.2)

M*N is the size of the image $f^{\wedge}(i,j)$ and f(i,j) are the matrix elements of original and compressed image .compression ratio is calculated as the ratio of non-zero entries in the original image to the non-zero entities in the compressed image.

CR= original image/compressed image

VII.VALIDATION AND TEST DATA

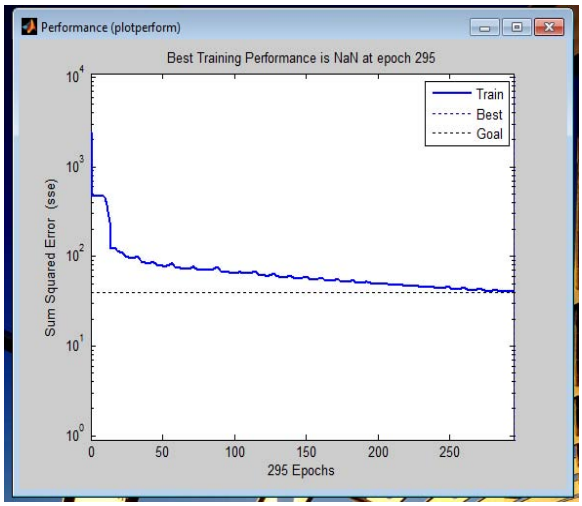

Figure 6.1: Graph for validation and Test data

In the below graph we are training the network based on three criteria's those are

1. TRAINING: These are presented to the network during the training and the network is adjusted according to its error.
2. VALIDATION: These are used to measure the network realization ion, and to halt training when generalization stops improving.
3. TESTING: These have no effect on training and so provide an independent measure of network performance during and after training.

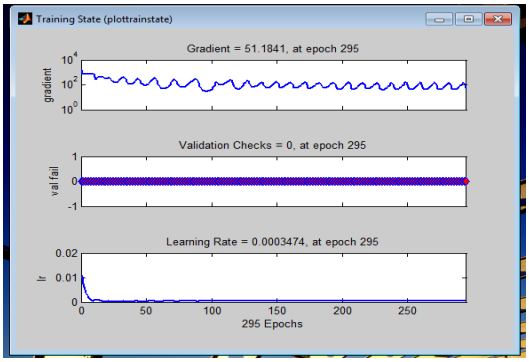

Figure 6.2: Graph for training the data

A.Regression Graph:

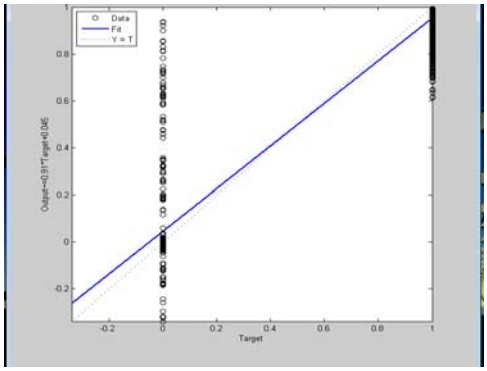

Figure 6.3: Regression graph

B.Regression:

Regression is predicting the output up to which training level we will get the accurate output.

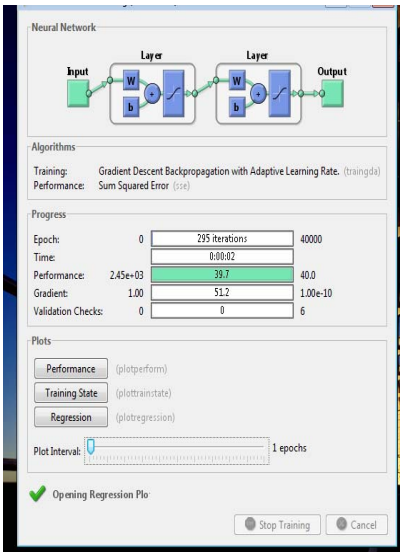

Fig 6.4:Training of data using Neural Networks

C.Final Outputs

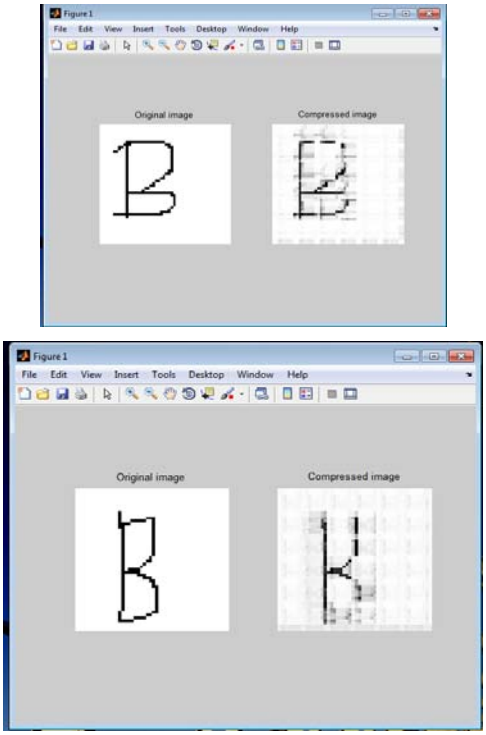

Figure 6.5: input image and the compressed image

CONCLUSION

The implementation of hand written symbol recognition using Back propagation algorithm in Neural Networks on image compression with good performance has been achieved and here considering the different attributes for image compression such as Root Mean Square Error and peak signal to noise ratio .The back propagation algorithm in Neural Networks has been tested for different data tests and observed that the convergence time for train the back propagation algorithm in Neural Networks is very fast.

FUTURE SCOPE

The field of Neural Networks is emerging technology in day to day and it requires more and more revolution and evolution in pattern recognition .Back propagation algorithm was successfully implemented for image processing .The same experiment also be implemented in other type of Neural Networks to see how the performance will increases with Back propagation algorithm

REFERENCES

[1] Ronald A .DeVore, Bjorn Jawerth, and Bradley J.Lucier. “Image compression through wavelet transforms coding” IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 38. NO .2, MARCH 1992.

[2] Abbas Razavi, Rutie Adar, Isaac Shenberg, Rafi Retter and Rami Friedlander “VLSI implementation of an image compression algorithm with a new bit rate control capability” Zoran Corporation, 1705 Wyatt Drive, Santa Clara, CA 95054.This paper was published in IEEE international conference on 26 th march 1992 on volume 5 and pages 669-672.

[3] P.Moravie, H.Essafi, C. Lambertt-Nebout and J-L. Basill. “Real time image compression using SIMD architecture” Centre Spatial de Toulouse 18 Avenue Edouard Belin BP 1421. This paper appears in computer architecture for machine perception on 20 September 1995 on pages 274-279.

[4] David Jeff Jackson and Sidney Jwl Hannah “Comparative analysis of image compression technique” Department of Electrical Engineering ,The University of Alabama, Tuscaloosa, AL 35487.This paper appears in system theory 1993,proceeding SSSST’93, twenty fifth edition southeastern symposium on 9 th march 1993 on pages 513-517.

[5] JJiang “Neural network technology for image compression” Bolton Institute, UK. This paper appears \in broadcasting convection 1995,IBC 95,on 18 September 1995 on pages 250-257.

[6] Michael T. Kurdziel. “Image compression and transmission for HF radio system”. Harrish corporation RF communication division Rochester, NY. This paper appears in MILLCON 2002, volume 2 on pages 1281- 1285 .

[7] Aaron T. Deever and Sheila S.“Lossless image compression with projection based and adaptive reversible interger wavelet transform”. This paper appears in IEEE transactions of image processing, volume 12, on May 2003

[8] Robert D. and Haykin S., “Neural Network Approaches to Image Compression” , Proc. Of I EEE, vol.83, pp. 288-303, Feb. 1995.

[9] Sicuranzi G.L., Ramponi G., and Marsi S., “Artificial Neural Network for Image Compression”, Electron Lett., vol. 26, pp. 477- 479, Mar. 1990.

[10] Dumitras, A. and Kossentini, F., “Fast and High Performance Image Subsampling using Feedforward Neural Networks," IEEE Transactions on Image Processing, vol. 9, no. 4, pp 720-728, Apr. 2000.

[11] Al-Alaoui, M. A, Mouci, R. and Mansour, M. "A RedundancyApproach to Classifier Training" , Proceedings of the 7th IEEEInternational Conference on Electronics Circuits and Systems (ICECS'2K),Vol. II, pp. 950-953, Sept. 2000.