

# Image Recognition using Neural Network

## An Application of Auto-associative Neural Network

Praveen Kumar

Dept. of Instrumentation and Control Engineering

Manipal Institute of Technology

Udupi, Karnataka, India

**Abstract**—A human being has the capacity to memorize a pattern and can also recall it. It is well-known that neural network is effective for classification problems. Auto associative neural network can be used to memorize the image by training the network. Auto-associative neural network can be used as memory and hence finds application in image recognition. The matrix associated with gray image can be used to find the associated weight matrix which is used to train the model network. After training the model it will recognize the image appropriately. It is noticed that the model developed is accurate enough to recognize the image even if the image is distorted or some portion of data is missing from the image. This model eliminates the long time consuming process of image recognition.

**Keywords**—Auto-associative neural network, weight matrix Image recognition

### I. INTRODUCTION

Soft computing is an emerging field built up of latest techniques like fuzzy logic, artificial neural network etc., these techniques are being used increasingly for complex data that involves vagueness and uncertainty. Artificial neural networks are inspired by biological neuron found in human brain. A neural net consist of a large number of simple processing elements called neurons or nodes. Each node is connected to other node by means of directed communication link each with an associated weight. The weights represent the information being used by the net to solve the problem. A neural network is characterized by

- **Architecture:** It is the pattern of connections between the nodes in the net. The connection can be of single layer type or multilayer type.
- **Learning Algorithm:** The method of determining the weights of the connecting links. The learning method can be supervised or unsupervised and in some cases they are fixed. [9]
- **Activation Function:** The inputs to the output node will be sum of all signal from each node multiplied with the corresponding weight over the connection link. The output is obtained by applying the activation function at the output node. Typical functions are sigmoidal, binary step, bipolar step and identity.[11]

Neural network finds application in various areas of image processing such as compression [6], recognition [7] [5] and segmentation [8].

In image recognition there are two principal arguments used one is vector and strings. The vectors are represented as a column matrix form of dimension  $n \times 1$ [10]. Image can be recognized based on minimum distance classification method [1] where euclidean distance is measured between two vectors, one of a given vector and other of vector stored. Other method is based on correlation [3], method based on statistical classification [2], method based on neural network [4].In this paper we use auto associative neural network to recognize a gray image represented in matrix form

### II. AUTO ASSOCIATIVE NEURAL NETWORK

The process of learning includes forming association between the related patterns. Memorization of a pattern (or a group of pattern) may be considered to be associating the pattern with itself. Associative neural nets are single layer nets for which weights are determined such that net stores the set of pattern associated. Each association is a pair of input-output vector s:t. If each target vector (t) is same as the sample vector(s), with which it is associated then the net is called as auto associative memory.

#### A. Architecture

Auto associative net has a single layer feed forward architecture as shown in Fig.1. The  $j^{\text{th}}$  output node is connected to the  $i^{\text{th}}$  input node using a connection link with weight  $w_{ij}$ .

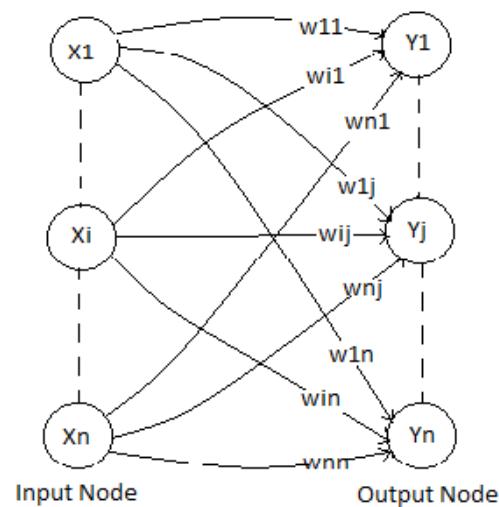


Fig.1. Architecture of auto associative neural network.

### B. Training the net

Training of the net is mainly to find the associated weight matrix or simply weights. The Hebb rule can be used to find the weights of the net as input and output are perfectly correlated. The stepwise procedure to train the net is given below.

Step 0: Initialize all the weights to zero

$$w_{ij} = 0 \text{ or } W = 0$$

W is weight matrix

Step1: To store the set of vector pair s: t, we find the weight matrix as the outer product of these vectors

$$S = (s_1, s_2, \dots, s_n)$$

$$T = (t_1, t_2, \dots, t_m)$$

$$W = S^*T$$

In general, for k different vectors the weight matrix can be given as

$$W = \sum_{p=1}^k s'(p) * t(p) \quad (1)$$

Step 2: The net input to the  $j^{\text{th}}$  output node from  $i^{\text{th}}$  input node will be

$$Y_{j\text{net}} = X_i * W \quad (2)$$

Step 3: The bipolar step function is applied as the activation function at the output node is shown in Fig.2.

$$Y = \begin{cases} 1 & Y_{j\text{net}} \geq 0 \\ -1 & Y_{j\text{net}} < 0 \end{cases} \quad (3)$$

### C. Sample Example

We try to illustrate an example to store a vector using auto associative network. Let vector  $[1 \ 1 \ 1 \ 1 \ -1 \ -1]$  to be stored.

$$S = [1 \ 1 \ 1 \ 1 \ -1 \ -1]$$

The weighted matrix has to be determined using hebbian rule to store an association s : t using (1)

$$W = S^*S$$

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 & -1 & -1 \\ -1 & -1 & -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & -1 & 1 & 1 \end{pmatrix}$$

Testing the net with the same vector as that is stored includes an vector at the input node to be  $X=S$ . The net input at the output node using (2)

$$Y_{\text{in}} = [6 \ 6 \ 6 \ 6 \ -6 \ -6]$$

Applying the binary step activation function as in (3) will yield the output vector as

$$Y = [1 \ 1 \ 1 \ 1 \ -1 \ -1]$$

Thus the net recognizes the vector X at the input node as the vector obtained at the output is same as the vector S that is stored.

Now, Let the input vector X has some data missing points is applied to the input node of the net.

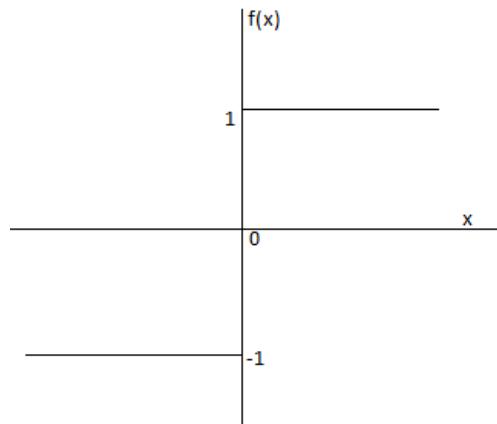


Fig.2.Binary step activation function.

$$X = [1 \ 1 \ 1 \ 0 \ -1 \ 0]$$

Then using (2) and (3) we get the output vector which has been stored.

$$Y_{\text{in}} = [4 \ 4 \ 4 \ 4 \ -4 \ -4]$$

$$Y = [1 \ 1 \ 1 \ 1 \ -1 \ -1]$$

Hence the net will recognize the input vector accurately even few data are missing.

### III. IMAGE RECOGNITION USING AUTO ASSOCIATIVE NET

Any gray image can be represented in the form of a matrix. The entries in the matrix can be converted into suitable form either into binary or bipolar representation for associative network. Each row vector is stored using a weight matrix then all such weight matrices for different row vectors are added to form final weight matrix. The true image matrix and image matrix with salt and pepper noise added is applied at the input nodes of the matrix to test the network.

#### A. Proposed Method

Firstly, we consider an image to be stored in the net Fig.3. The image matrix is reconstructed into a square matrix by considering suitable part of image. In our discussion we have considered the matrix of order 128 by 128. It is important to convert the entries of the matrix into bipolar values as the auto associative network works with binary or bipolar data.

We designate the image matrix in bipolar form to be used to train the net as S and associated weight matrix as W which is formed by taking outer product of the row vector of S with itself and summing all such outer product matrices. The net input to the output node is calculated using (2). Finally the bipolar step activation function is applied at the output node (3). We test the net with different inputs at the input node.

#### IV. EXPERIMENTAL RESULTS

The training and testing for the neural net to recognize an image is carried out on MATLAB platform. Any input image at the input node of the net is given by the matrix  $X$ . If the output matrix  $Y$  is same as the stored matrix  $S$  then the image is said to be recognized. As we compare two matrixes the number of position or pixel by which they differ can be obtained.

The grey image of trees shown in Fig.3. is used to train the net. The image is originally of the order 258 by 350 which is reconstructed to a square matrix of size 128 by 128. The pixels of the square matrix formed are converted to bipolar representation. The weight matrix  $W$  is obtained using (1)

##### A. Testing with same input as stored

If the input image  $X$  is same as the stored image  $S$  then the net input to the output node is  $X*W$ . After applying the binary step activation function we get output matrix  $Y$ . The stored matrix  $S$  and output matrix  $Y$  are compared, the number of locations by which the two matrix differ will give the number of failure occurred in this case there will be no failure hence image is recognized successfully.



Fig.3.Trees image to be stored in the network



Fig.4.Trees image with salt and pepper noise of density 0.08

##### B. Testing with distorted input image

The actual image is being distorted by adding salt and pepper noise to it hence the input image  $X$  is slightly different from the stored image  $S$  as shown in Fig.2. Again the net input  $X*W$  is calculated and activation

function is applied at the output node to obtain output matrix  $Y$ . The number of failures depends on the amount by which image is being distorted which is determined by salt and pepper noise. Table I shows the number of failures for different noise levels. If the number of failure is zero then image will be recognized successfully

TABLE I. NUMBER OF FAILURES WITH NOISE

Salt and Pepper noise density	0.001	0.004	0.007	0.008	0.01
Number of failures	0	78	94	96	104

#### V. CONCLUSION AND FUTURE WORK

This paper proposes the application of auto associative neural net to recognize an image. The experimental results show that the model recognizes the image with few data missing or small distortion. The model developed is computationally inexpensive, simple along with good recognition rate.

The auto associative net requires the vectors to be orthogonal to store using the same weight matrix which limits the model from storing more images using same weight matrix. In future the modification to the model can be done to store more images using same weight matrix.

#### REFERENCES

- [1] Yanhua Ma and Chuanjun Lia,"A Recognition algorithm for Chinese character based on minimum distance classifier", 2<sup>nd</sup> Intl. workshop on computer science and engineering,2009
- [2] Liang Wang and Tieniu Tan,"Automatic Gait Recognition based on Statistical Shape Analysis", IEEE Trans. On Image Processing,vol.12,No.9, September 2003
- [3] Lihang Zhao,Yu Cai and Xinhe Xu,"Face Recognition based on Correlation of Wavelet Transform Images", Proceedings of 6<sup>th</sup> world congress on Intelligent Control and Automation, June 21-23,2006,China
- [4] Gaurav Kumar and Pradeep Kumar Bhatia," Neural Network based approach for Recognition of Text Images", IJCA vol.62,No-14,January 2013
- [5] A.Rajavelu,M.T.Mausai and M.V. Shrivaika,"A Neural Network approach to character recognition." Neural Networks,vol.2 1989 pp 387-393
- [6] Suprava Patnaik and R.N.Pal,"Image compression using Auto associative Neural network and Embedded zero tree coding",3<sup>rd</sup> IEEE signal processing workshop, March 20-23,2001, Taiwan.
- [7] Indira.S.U and Ramesh.A.C,"Image segmentation using neural network and generic algorithm: A comparative analysis", Intl. Conference on Process Automation, Control and Computing(PACC) ,2011
- [8] Md.Iqbal Quraishi,J Pal Choudhary and Mallika De,"Image Recognition and processing using Artificial Neural Network",1<sup>st</sup> Intl. conference on Recent Advances in Information Technology (RAIT),2012
- [9] Simon Haykin, Neural Network A Comprehensive foundation, 2<sup>nd</sup> ed, 1998, pp.750-768.
- [10] Rafael.C.Gonzalez and Richard.E.Woods, Digital Image Processing,2<sup>nd</sup> ed,2002,pp. 712-730
- [11] Laurene Fausett,Fundamentals of Neural network Architecture, Algorithm and Application,3<sup>rd</sup> ed,2008 pp.11-20 and pp. 121-125