Volume 2, Issue 8, August 2012 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

ANN Implementation for Reconstruction of Noisy Numeral Corrupted By Speckle Noise

Smita K. Chaudhari*, G.A.Kulkarni
Dept. Of Electronics & Communication Engg
SSGBCOET,Bhusawal
smita.c20@gmail.com, girish227252@rediffmail.com

***Abstract*— Neural Network (NN) is information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. Neural Networks are known to be capable of providing good recognition rate in presence of noise. Neural Network with various architectures and Training algorithms have successfully been applied for letter or character recognition [1]. This paper uses hamming netwok and Hopfield network to recognize noisy numerals. The recognition results of the noisy numeral showed that the network could recognize normal numerals with 100% accuracy, numerals added with speckle noise at average of 91%.**

***Keywords*— Character Recognition, Hamming Network, Noisy Numeral, Speckle Noise, Hopfield Network**

I. INTRODUCTION

Recently, neural network becomes more popular as a technique to perform character recognition. It has been reported that neural networks could produce high recognition accuracy. Neural networks are capable of providing good recognition at the present of noise that other methods normally fail.

An Artificial Neural Network (ANN) is information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This is true of ANNs as well. Neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyse.

Neural networks take a different approach to problem solving than that of conventional computers. Conventional computers use an algorithmic approach i.e. the computer follows a set of instructions in order to solve a problem. Unless the specific steps that the computer needs to follow are known the computer cannot solve the problem. That restricts the problem solving capability of conventional computers to problems that we already understand and know how to solve. But computers would be so much more useful if they could do things that we don't exactly know how to do. Neural networks process information in a similar way the human brain does.

Object recognition is the study of how machines can observe the environment, learn to distinguish patterns of interest and make reasonable decisions about the categories of patterns. The performance of a machine may be better than the performance of a human in a noisy environment due to the factors: human performance degrades with increasing number of targets; where as the performance of a machine does not depend on the size of the set of targets. The performance of a machine does not degrade due to fatigue caused by prolonged effort. A knowledge based system is desirable for reliable, quick and accurate recognition of objects from noisy and partial input images [3].

The McCulloch and Pitts model was utilized in the development of the first artificial neural network by Rosenblatt in 1959 [11]. This network was based on a unit called the perceptron, which produces an output scaled as 1 or -1 depending upon the weighted, linear combination of inputs.

The optical character recognition system for hand printed numerals of noisy and low-resolution measurement consists of the two-stage feature extraction process. In the first stage a set of primary features insensitive to the quality and format of a black-white bit pattern are extracted. In the second stage, a set of properties capable of discriminating the character classes is derived from primary features. The system is simple and reliable in that only three kinds of primary features are needed to be detected. The recognition is based on the decision tree which tests the logic statements of secondary features. [12]

The importance of using a hierarchical network is shown in literature [16] Seong-Whan Lee finds a new scheme for off-line recognition of totally unconstrained handwritten numerals using a simple multilayer cluster neural network trained with the back propagation algorithm which avoids the problem of finding local minima & improves the recognition rates [10]

H. K. Kwan introduced multilayer recurrent neural networks in the form of 3-layer bidirectional symmetrical and asymmetrical associative memories are presented. The networks possess the features of both a multilayer feedforward neural network and a bidirectional associative memory. These networks can have two modes of recalling, namely, recalling by one pattern and recalling by a pattern pair in[12]

Recognition of Noisy Numerals using Neural Network by Mohd Yusoff Mashor and Siti Noraini Sulaiman.This paper uses MLP network trained using Levenberg-Marquardt algorithm to recognise noisy numerals. The recognition results of the noisy numeral showed that the network could recognize normal numerals, blended numerals[15].

II. BACKGROUND & TERMINOLOGY.

The Hamming network method was developed by a mathematician, Richard W. Hamming. He has many contributions not only in the mathematical field, but also for computer science and telecommunication [5]. He was also the founder and has been the president of Association for Computing Machinery. Hamming network method is developed to solve pattern recognition problems which use binary format, such as a matrix with only two possible values, 0 and 1. In the Hamming network, there is a matrix which stores the patterns of all objects, called the prototype data matrix. The patterns will not be learned by the system, but rather to be stored as a matrix data. The matrix will be used to define the output of the network. The objective of the Hamming network is to decide which prototype matrix is closest to the input matrix. It calculates the similarities between the prototype matrix of all objects and the input.

It is designed explicitly to solve binary pattern recognition problems. It has both feed forward and recurrent layer. The number of neuron in the first layer is the same as the number of neurons in the second layer. The objective of the hamming network is to decide which prototype vector is closest to the input vector. This decision is indicated by the output of the recurrent layer. When the network converges, there will be only one nonzero output. This indicates the prototype pattern that is closest to the input vector.

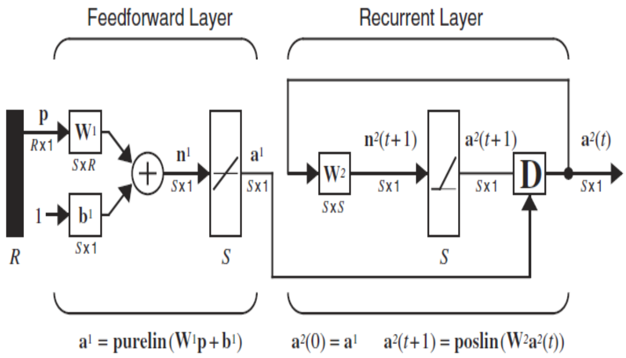

Fig.1 Hamming Network

A. Feedforward Layer

Feedforward layer is a layer which calculates the correlation between each patterns of the prototype matrix and the input matrix (figure 1). The calculation results will be processed to generate the output neurons for this layer.

Feedforward layer is a layer which calculates the correlation between each patterns of the prototype matrix and the input matrix (figure 1). The calculation results will be processed to generate the output neurons for this layer.

As shown in the figure 1, the layer has the input matrix from p, which has the dimension as R x 1. This input matrix goes to the weight matrix (W1) with the dimension of S x R. The net of this layer (n1) will be the sum of the W1p and the bias input b. The weight matrix of W1 will be the matrix of the prototype data which include the patterns of all objects. The element of the bias b will be given as the number of R. The transfer function which is used in this layer is the linear transfer function (purelin). This function will not change the value so the output of this feedforward layer (a1) will be given as: a1 = purelin(W1p + b1) The output neurons of this layer will be used as the initial input for the next layer, the recurrent layer.

B.Recurrent Layer

The recurrent layer is also known as a competitive layer. In this layer, there is a neuron for each prototype pattern. The neurons in this layer are initialized with the output neurons of the feedforward layer, which indicate the correlation between the prototype patterns and the input matrix.

The neurons will compete each other to determine a winner. When the processes are finished, there will be only one neuron with nonzero output. This neuron indicates the prototype pattern that is closest to the input.

The processes in the recurrent layer will be divided into iterations. After one iteration is finished, a function will check whether there is only one nonzero output. If so, the process in this layer will be stopped, and the process will continue to generate the output.

In Figure D is the function to check whether there is only one nonzero output. W2 is the weight matrix for this layer with the dimension of S x S. The iteration number will be given as t, and it will be added by one until the iteration stopped. The activation function which is used is the positive linear transfer function (poslin).
This function is linear for positive values and zero for negative values.

C. Speckle Noise

Speckle noise [Ga99] is a multiplicative noise. This type of noise occurs in almost all coherent imaging systems such as laser, acoustics and SAR(Synthetic Aperture Radar) imagery. The source of this noise is attributed to random interference between the coherent returns. Fully developed speckle noise has the characteristic of multiplicative noise. Speckle noise follows a gamma distribution and is given as

$$F(g) = \frac{g^{\alpha-1}}{(\alpha-1)!a^{\alpha}} e^{-\frac{g}{a}},$$

where variance is $a2\alpha$ and g is the gray level.
On an image, speckle noise (with variance 0.05) looks as shown in Image 3[Im01]. The gamma distribution is given below in Figure 2.

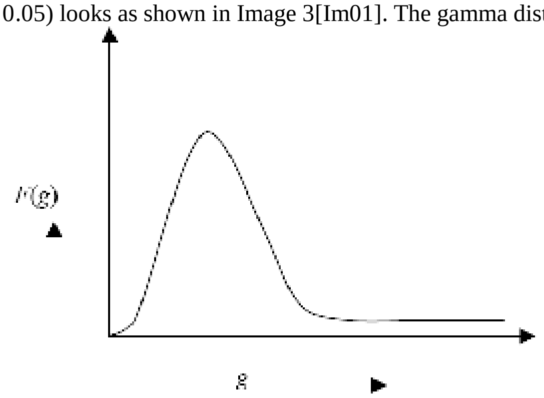

Fig 2.Gamma Distribution

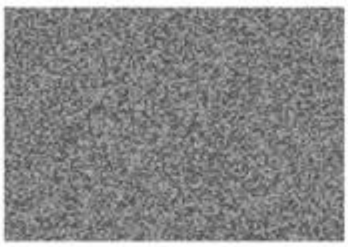

Fig 3.Speckle Noise.

If the multiplicative noise is added in the image, speckle noise is a ubiquitous artifact that limits the interpretation of optical coherence of remote sensing image. The distribution noise can be expressed by:

$$J = I + n*I \quad (3)$$

Where, J is the distribution speckle noise image, I is the input image and n is the uniform noise image by mean o and variance v.

III. DESIGN AND IMPLEMENTATION OF THE SYSTEM.

The system designed in this paper associates every fundamental pattern with itself. That is, when presented with $\mathbf{x}i$ as input, the system should produce $\mathbf{x}i$ at the output. In addition, when presented with a *noisy* (corrupted) version of $\mathbf{x}i$ at the input, the system should also produce $\mathbf{x}i$ at the output. The system which is developed is a system that gets an input of digit, process it through the network, and generates the result. The digit which are used in the development are limited to printed digit from 1 to 9. The system has some prototype data that consists of the pattern of digits, from 1 to 9. This prototype data is used as the weight matrix for the process in the feedforward layer of the Hamming network. The system is built using the MATLAB and the images are processed using the Microsoft Paint.

The type of the image file is bitmap (.bmp) . The image is read & converted into 64×64 matrix form. This matrix is converted to 8×8 matrix to reduce the computations. Since two dimensional input can't be given to neural network then it is converted to 64×1 column vector and this column vector is the prototype pattern The system will have a function that simulates the Hamming network. The function will act as the network and process the input data to generate the output. The output neuron of the network indicates the result of the recognition process.

In the feedforward layer, **p** is the input matrix. It will be the matrix of the input image which size is 8 x 8. Therefore, the input **p** will be a matrix of 64 x 1. The **R** number is 64, which is the number of input neuron, while **S** is the number of the output neuron for this network, which is 9. The weight matrix **W1** will be generated using the prototype data. It will take the prototype data matrix of the 9 digits, so the weight matrix will be a matrix of 9 x 64. The bias **b** will be a matrix of 9 x 1.

In the recurrent layer, there are 9 output neurons which represent the number of digits.

The Speckle noise having different density is added in the image by using the MATLAB function & then it is processed & recognized by using the designed system.

IV. PERFORMANCE ANALYSIS.

The task was to design a system which associates every fundamental pattern with itself. That is, when presented with $\mathbf{x}^i$ as input, the system should produce $\mathbf{x}^i$ at the output. In addition, when presented with a *noisy* (corrupted) version of $\mathbf{x}^i$ at the input, the system should also produce $\mathbf{x}^i$ at the output.

Let the Hamming distance between two binary vectors x and y (of the same dimension) be denoted as d(x, y). The design phase of the Hamming memory involves simply storing all the patterns of the fundamental memory set. In the recall phase, for a given input memory key $\mathbf{x} \in (\mathbf{0}, 1)^N$, the retrieved pattern is obtained as follows

(1) Compute the Hamming distances $dk = d(x, \mathbf{x}k)$, $k = 1, 2, \ldots\ldots, m$.
(2) Select the minimum such distance $dk = \min \{d1, d2, \ldots..dm\}$
(3) Output the fundamental memory $y = \mathbf{x}^k$ (closest match)
(4) Input: storage patterns for Hamming network.
(5) Input prototype images for digits 1-9 from .bmp format.
(6) Example: p = 64*64 matrix of prototype input image of digit 1.

Fig. 4. Prototype Images

Scale data and display as image to use the full colormap. Colormap (gray) sets the current figure's colormap to gray. The values are in the range from 0 to 1. A colormap matrix may have any number of rows, but it must have exactly 3 columns. Each row is interpreted as a color, with the first element specifying the intensity of Red light, the second Green light, and the third Blue. Color intensity can be specified on the interval 0.0 to 1.0. For example, [0 0 0] is black, [1 1 1] is white, [1 0 0] is pure Red, [.5 .5 .5] is gray, and [127/255 1 212/255] is aquamarine. Resizes a matrix map image to an 8*8 matrix to reduce computations. i.e. Convert and compression of image.

Example: p2 = resize (p, [8, 8]).

Table 4.1 Reconstruction Efficiency/ Recognized Output Digit for Speckle Noise.

Input Digit	Noise density	Recognized output	% Accuracy	Iteration
1	1	1	77	10
	5	1		15
	10	1		24
	15	1		13
	20	1		33
	25	1		10
	30	1		15
	35	9		21
	40	1		18
	45	1		10
	50	6		17
	55	1		13
	60	6		24
	70	1		12
	75	1		15
	80	9		21
	90	1		19
	100	1		10

Input Digit	Noise density	Recognized output	% Accuracy	Iteration
2	1	2	100	8
	5	2		23
	10	2		8
	15	2		12
	20	2		10
	25	2		13
	30	2		10
	35	2		11
	40	2		16
	45	2		20
	50	2		11
	55	2		20
	60	2		9
	70	2		22
	75	2		9
	80	2		12
	90	2		10
	100	2		17
3	1	3	88	17
	5	3		12
	10	3		15
	15	3		17
	20	3		10
	25	3		12
	30	3		26
	35	2		20
	40	3		12
	45	None		64
	50	3		33
	55	3		11
	60	3		21
	70	3		13
	75	8		20
	80	3		10
	90	3		11
	100	3		21
4	1	4	61	9
	5	4		9
	10	4		19
	15	4		14
	20	4		18
	25	4		10
	30	4		15
	35	4		17
	40	8		22
	45	6		8
	50	6		19
	55	6		10
	60	8		20
	70	8		31
	75	4		14
	80	2		64
	90	4		25
	100	4		8

Input Digit	Noise density	Recognized output	% Accuracy	Iteration
5	1	5	94	8
	5	5		16
	10	5		18
	15	5		25
	20	5		14
	25	5		22
	30	5		25
	35	5		8
	40	5		15
	45	5		8
	50	8		48
	55	5		7
	60	5		9
	70	5		10
	75	5		11
	80	5		13
	90	5		15
	100	5		19
6	1	6	100	9
	5	6		11
	10	6		10
	15	6		9
	20	6		11
	25	6		11
	30	6		12
	35	6		12
	40	6		11
	45	6		10
	50	6		8
	55	6		11
	60	6		12
	70	6		12
	75	6		9
	80	6		13
	90	6		12
	100	6		14
7	1	7	94	6
	5	7		12
	10	7		6
	15	7		10
	20	7		40
	25	7		12
	30	7		13
	35	7		14
	40	7		13
	45	7		23
	50	7		11
	55	7		38
	60	7		25
	70	7		8
	75	7		13
	80	9		34
	90	7		31
	100	7		8

Input Digit	Noise density	Recognized output	% Accuracy	Iteration
8	1	8	100	5
	5	8		7
	10	8		8
	15	8		11
	20	8		13
	25	8		9
	30	8		10
	35	8		11
	40	8		6
	45	8		6
	50	8		10
	55	8		7
	60	8		15
	70	8		10
	75	8		8
	80	8		8
	90	8		10
	100	8		12
9	1	9	100	10
	5	9		10
	10	9		8
	15	9		14
	20	9		8
	25	9		12
	30	9		10
	35	9		9
	40	9		12
	45	9		10
	50	9		12
	55	9		7
	60	9		11
	70	9		9
	75	9		12
	80	9		20
	90	9		15
	100	9		12

Table 4.1 shows that there is different accuracy and iterations for each digit from 1 to 9. Digit 2,6,8,9 have 100% accuracy, digit 5 and 7 have 94% accuracy, digit 3 have 88% accuracy, digit 1 have 77% accuracy and digit 4 have 61%.

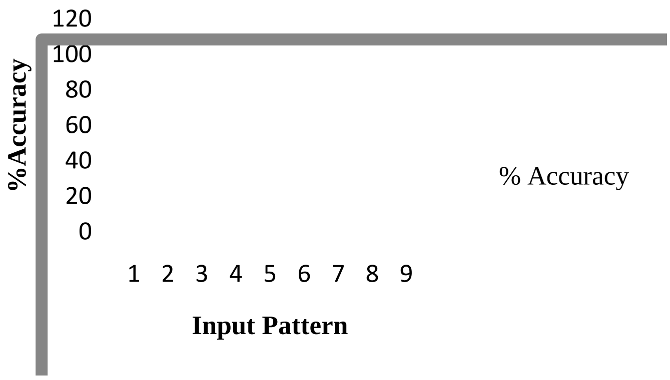

Fig.5. Reconstruction Efficiency for Speckle Noise

Figure 5 shows graph of input pattern verses accuracy

V. CONCLUSION.

Pattern recognition can be done both in normal computers and neural networks. Computers use conventional arithmetic algorithms to detect whether the given pattern matches an existing one. It is a straightforward method. It will say either yes or no. It does not tolerate noisy patterns. On the other hand, neural networks can tolerate noise and, if trained properly, will respond correctly for unknown patterns. Neural networks will not perform miracles, but if constructed with the proper architecture and trained correctly with good data, they will give amazing results, not only in pattern recognition but also in other scientific and commercial applications.

Two models hamming and Hopfield image pattern classification and reconstruction, these two algorithms is to supply the prototype images in the model memory and then use the memory later to identify the stored patterns; when partial input is given as input to the model . Efficiency of both models varies according to the noise. For speckle noise it gives average accuracy 91%. Because of the iterative nature of the two algorithms; software implementation of these two algorithms is simple and efficient.

REFERENCES

[1] Hagan M.T., Demuth H. B., Beale M., "*Neural Network Design*", Thomson Learning Vikas Publishing House, 1996.

[2] R C. Gonzalez, R. E. Woods, "*Digital Image Processing*", Pearson Education, Inc. and Dorling Kindersley Publishing, Inc.2008.

[3] S Jayaraman, S Esakkirajan, T Veerakumar, "*Digital Image Processing*", Tata McGraw Hill Education, 2009

[4] Earl, Gose, Richard Johnsonbaugh, Steve Jost, "*Pattern recognition and Image analysis*",Asoke K. Ghosh, Prentice Hall, 1997.

[5] "Statistical Pattern Recognition: A Review", Anil K. Jain, Fellow, IEEE, Robert P.W.Duin, and Jianchang Mao, Senior Member, *IEEE Transactions On Pattern Analysis And Machine Intelligence*, Vol. 22, No. 1, January 2000.

[6] Zurada, J.M., "*Introduction to Artificial Neural Systems*", Jaico publishing House, Mumbai, 2002.

[7] 2002) The IEEE website. [Online]. Available: http://www.ieee.org/ "PDCA12-70 data sheet," Opto Speed SA, Mezzovico, Switzerland.

[8] McCulloch, W.S., and Pitts, W. (1943), "A Logical Calculus of the Ideas Immanent in Nervous Activity," Bulletin of Mathematical Biophysics, 5, 115-133.

[9] Brion IC. Dolenko and Howard C. Card, "Handwritten Digit Feature Extraction and Classification Using Neural Networks", '*CCECE',* 1993,IEEE 0-7803-1443, PP 88-91.

[10] Seong-Whan Lee, " off-line recognition of totally unconstrained handwritten numerals using a simple multilayer cluster neural network", *IEEE transactions on pattern analysis and machine intelligence*, vol. 18, no. 6, june 1996 , 648-652

[11] Rosenblatt, F. (1959), "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain," *Psychological Review* 65:386-408.

[12] "Recognition of handprinted numerals by two-stage feature extraction", *IEEE transactions on systems science and cybernetics*, april 1970

[13] *"*Handwritten alphabet recognition using hamming network", Arnold Aribowo, Samuel Lukas, Handy, *Seminar Nasional Aplikasi Teknologi Informasion*, 2007 (SNATI 2007) ISSN: 1907-5022 Yogyakarta, 16 June 2007

[14] A Two-Level Hamming Network for High Performance Associative Memory", by Nobuhiko Ikeda, Paul Watta, Metin Artiklar and Mohamad H. Hassoun.

[15] "Recognition of Noisy Numerals using Neural Network", Mohd Yusoff Mashor and Sitz Noraini Sulaiman Centre for Electronic Intelligent System (CELIS), School of Electrical and Electronic Engineering, University Sains Malaysia Engineering Campus, 14300 Nibong Tebal Pulau Pinang, Malaysia.

[16] Amit Choudhary, Rahul Rishi, Savita Ahlawat, "Performance Analysis of Feed Forward MLP with various Activation Functions for Handwritten Numerals Recognition" *IEEE*, Volume 5, 2010,pp 852-856,

[17] Stuti Asthana, Farha Haneef, Rakesh K Bhujade, "Handwritten Multiscript Numeral Recognition using Artificial Neural Networks*" IJSCE 2231-2307,* Volume-1, Issue-1, March 2011 pp 1-5

