Development and Optimizing of a Neural Network for Offline Signature Recognition

Sunil Kumar, Jitender Khurana, Kamal Sardana and Naveen Goel

Abstract:-The signature recognition work has been actively researched in recent years. Many biometric techniques have been proposed by researchers for personal identification in the past. Among them signature recognition is the most widely known non-vision based technique. As we know that signatures continue to play an important role in financial, commercial and legal transactions. A signature by authorized person is considered to be the "seal of approval". There are various techniques to signature recognition with a lot of scope of research. In this paper off-line signature recognition and verification system using Artificial Neural Network (ANN) is purposed. The purposed network based upon the adaption of ANN to recognized signature to connected type pattern. The purpose ANN was trained with back propagation with momentum and adaptive learning rate.

A triple hidden layer ANN with 100 inputs 58.38.20 hidden neurons layers and 5 neurons in output layers gives best results as compared with other networks.

The network is for

Keywords:-Off line signatures, Artificial Neural Network (ANN), Error Back Propagation.

I. INTRODUCTION

Signatures themselves are worthless but because they are used as authorization of various things like cheques and wills, they become targets of forgers. Computer detection of forgeries may be divided into two classes, the on-line approach and the off-line approach. The on-line approach of signature verification involves the use of dynamic information which is attained during the actual signing of the signature. This information includes the velocity of the pen, the pressure exerted during the signing process and the acceleration of the pen. The off-line approach uses only the final result of the signing process, i.e. the image of the signature [1].

In this paper our main focus is on the offline signature recognition by using ANN. An ANN model of visual pattern recognition, called the Necognitron, was previously proposed by Kunihiko Fukushha [2, 3]. The ability of ANN to learn and approximate relationships between input and output is decoupled from the size and complexity of the problem [4]. By using this ability we developed a neural network for off line signature recognition. The functioning of neural network is divided in to two phases:

1 Training or Learning Phase
2 Testing Phase

Sunil Kumar[1], Jitender Khurana[2], Kamal Sardana[3], Naveen Goel[4]
1 M.Tech Student, Baba Mastnath Engineering College, Rohtak, INDIA
2 Associate Professor, Baba Mastnath Engineering College, Rohtak, INDIA
3 Assistant Professor, The Technology Institute of Textile & Sciences, Bhiwani, INDIA
4 Associate Professor, Vaish College of Engineering, Rohtak, INDIA
(Email: 1. sunilkumar_rtk@rediffmail.com, 2. jitukhurana@gmail.com, 3.sardanakamal@yahoo.com, 4.ng4u1@rediffmail.com)

In the training phase the system is providing the information regarding the patterns. The patterns are provided in the form of 0's and 1's.As the neuron process each bit converted in to weight. In the testing phase, the system is provided with testing patterns that has to be recognized by the system. The system generates weights for these patterns and compares these weights with the pattern in its database. If it recognizes the test pattern, then it gives the corresponding output.

The neural network can be optimized with respect to various parameters, like number of hidden layers used, number of nodes in the hidden layers, learning rate (L.R.) and momentum rate (M.R.) etc. to achieve minimum error in the learning and testing. The signature recognition has been done by using optimum neural network structure.

II. SIGNATURE RECOGNITION MODEL USING ANN

The signature recognition by using neural network approach in done in three separate steps [5]. In the first step the give signature is converted in to binary data form. This is done by grid method. Here we use 10*10 matrixes for each signature. The matrix used is given in fig.1:

1,1	1,2	1,3	1,4	1,5	1,6	1,7	1,8	1,9	1,10
2,1	2,2	2,3	2,4	2,5	2,6	2,7	2,8	2,9	2,10
3,1	3,2	3,3	3,4	3,5	3,6	3,7	3,8	3,9	3,10
4,1	4,2	4,3	4,4	4,5	4,6	4,7	4,8	4,9	4,10
5,1	5,2	5,3	5,4	5,5	5,6	5,7	5,8	5,9	5,10
6,1	6,2	6,3	6,4	6,5	6,6	6,7	6,8	6,9	6,10
7,1	7,2	7,3	7,4	7,5	7,6	7,7	7,8	7,9	7,10
8,1	8,2	8,3	8,4	8,5	8,6	8,7	8,8	8,9	8,10
9,1	9,2	9,3	9,4	9,5	9,6	9,7	9,8	9,9	9,10
10,1	10,2	10,3	10,4	10,5	10,6	10,7	10,8	10,9	10,10

Fig 1: 10*10 Matrix for signature recognition

In the second step took the output of first step and trained the neural network. Training should take a few minutes, and it is completed after about 140-240 training cycles.

By changing the number of input layer neurons and hidden layer neurons the number of training cycles may increase or decrease [6]. Too few neurons and the network will be unable to learn anything. Too many neurons, and over learning is there. In the final or third step took the output of second step and create a network.

The multilayer perceptron with back propagation algorithm is a very effective topology of neural network for signature recognition [7, 8]. The basic neural network architecture, which has been used, is shown in fig. 2:

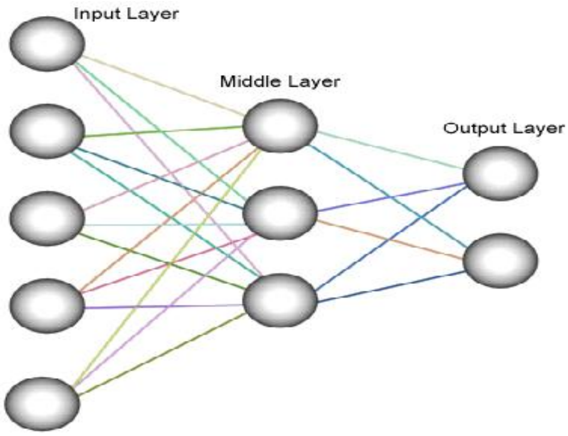

Fig 2: Basic Neural Network

The network has 100 (10*10 matrix for each signature) neurons in the input layer, one of each input. The input to the network is bit pattern taken in binary viz, 0 or 1.To generate a pattern of zero'& one's the small square of rectangular matrix, which is crossed by the signature line, taken as one, otherwise zero.

The matrix here we used is 10*10, hence No. of input is 100, as the No. Of signature to be recognized of 5 persons, hence No. Of nodes in output layer is 05. Here we use double hidden layer and triple hidden layer models of neural network. To find the optimum topology for signature recognition set optimum No. of hidden layer, No. of nodes in each hidden layer and value of Learning rate & momentum Rate.

III. SIGNATURE DATABASE

The signature samples were acquired from 05 individuals. Each individual was asked to give your signature on 10*10 rectangular matrixes. A total of 14 signatures were collected from each person to have samples of intra-personal variations. All these sample signatures were converted in 0's &1's and stored as genuine signatures. 02 samples for each Person was collected as test signatures, converted and stored as test samples. Thus the database used for testing the proposed system consists of 70 (14 x 05) genuine samples and 10 (2 x 5) test samples. The size of the database is 80 samples. [2]

VI. OPERATION PROCEDURE

The network is used to recognize the signature using the following procedure:

1. Apply the input vector to the network input.
2. Calculated the weighted sun NET of these inputs of each neuron in the hidden layer.
3. Apply activation function to these NET values to get out of hidden layer, which will be the input to the next layer.
4. With these values repeat steps 2 & 3.

The activation function, which is used in this model, is [1/ {1 + Exp. (-NET)}], where NET is the weighted sum of the inputs of the neurons. This activation function compresses the range of NET values [0, 1]. Thus each output values of the output layer shows the possibility of the input pattern to match the corresponding pattern.

As we have used Easy NN plus simulator for implementing the back propagation algorithm which perform all steps internally by itself. The weighted values are adjusted by the simulator itself, in response of error signal and adjusted till the average error is observed.

V. RESULTS

A. Recognition Result of Triple Hidden layer networks:

The purposed system for offline signature recognition was trained and tested on a database of total 80 signatures. In which 70 signatures are used for network training phase and remaining 10 for testing phase. To find an optimized network for offline signature recognition a hit and trail method is used because there is no formula for set perfect network parameters like no. of hidden layers, nodes in hidden layers, learning and momentum rate.

The values of maximum error, minimum error, average error, learning rate and momentum rate for different no. of hidden layer nodes taken in to analysis. The average error was fixed to 0.005. It is very much clear from table 1 that optimum results are obtained when no. of hidden layers are 60.41.22 (i.e. 60 in first hidden layer, 41 in second hidden layer and 22 in third hidden layer) and learning and momentum rate are 0.6 and 0.6 respectively. Using this network all the 10 signatures is correctly recognized. As the learning rate changed from 0.6 to 0.8, the result accuracy decreases from 10 to 8. The same thing happens when we change the momentum rate from 0.6 to 0.8. The table I shows the result of triple hidden layer network:

B. Recognition Result of double Hidden layer networks:

For the double hidden layer we take target error equal to 0.005 to train the network. In this case training of all networks is possible with best network, having layers 45 and 20 with 0.6 learning rate and 0.6 momentum rate. Results are shown in table II.

From table, it is clear that by varying the hidden layers from table I to II the results are changing quit dramatically. If we use two hidden layers first one having 44 nodes and second 21 nodes then the result decreases from 07 signatures are recognized out of 10 which is quite low. So the best results are found only when we use triple layers for network generation.

TABLE I
SIGNATURE RECOGNATION RESULT

H.L.	L.R.	M.R.	T.E.	Max. Er.	Min. Er.	No. Of Cycles	No. Of sign. Recog.
59.40.20	0.6	0.6	0.005	0.008516	0.002261	111	8
59.40.21	0.6	0.6	0.005	0.007240	0.002703	144	9
59.40.22	0.6	0.6	0.005	0.010844	0.002038	210	9
59.41.20	0.6	0.6	0.005	0.008938	0.002213	179	9
59.41.21	0.6	0.6	0.005	0.007442	0.003264	193	8
59.41.22	0.6	0.6	0.005	0.200598	0.000143	42500	Over Learning
60.38.20	0.6	0.6	0.005	0.008806	0.002648	161	9
60.38.21	0.6	0.6	0.005	0.009610	0.002419	151	9
60.38.22	0.6	0.6	0.005	0.008919	0.002245	148	8
60.39.20	0.6	0.6	0.005	0.009180	0.002180	185	8
60.39.21	0.6	0.6	0.005	0.009634	0.002140	149	7
60.39.22	0.6	0.6	0.005	0.009623	0.002806	176	7
60.40.20	0.6	0.6	0.005	0.008467	0.002335	170	9
60.40.21	0.6	0.6	0.005	0.008222	0.002417	153	8
60.40.22	0.6	0.6	0.005	0.010782	0.002118	148	9
60.41.20	0.6	0.6	0.005	0.008071	0.002885	163	9
60.41.21	0.6	0.6	0.005	0.012815	0.000444	2010	8
60.41.22	0.6	0.6	0.005	0.020144	0.002621	121	10
60.41.22	0.6	0.8	0.005	0.004228	0.000717	757	8
60.41.22	0.8	0.6	0.005	0.003548	0.001334	531	9
61.38.20	0.6	0.6	0.005	0.008538	0.002849	182	8
61.38.21	0.6	0.6	0.005	0.007291	0.002932	159	7

TABLE II
SIGNATURE RECOGNITION RESULT

H.L.	L.R.	M.R.	T.E.	Max. Er.	Min. Er.	No. Of Cycles	No. Of sign. Recog.
44.20	0.6	0.6	0.005	0.008882	0.001589	149	8
44.21	0.6	0.6	0.005	0.009411	0.001883	140	7
44.22	0.6	0.6	0.005	0.008633	0.001221	146	8
45.20	0.6	0.6	0.005	0.009872	0.001968	147	9
45.20	0.6	0.8	0.005	0.005695	0.000749	341	7
45.20	0.8	0.6	0.005	0.003575	0.000930	649	7
45.21	0.6	0.6	0.005	0.009595	0.002070	163	7
45.22	0.6	0.6	0.005	0.010890	0.001739	143	6
46.20	0.6	0.6	0.005	0.008430	0.002220	156	7
46.21	0.6	0.6	0.005	0.007987	0.001895	130	8
46.22	0.6	0.6	0.005	0.008497	0.002088	161	8
47.20	0.6	0.6	0.005	0.008347	0.002470	165	8
47.21	0.6	0.6	0.005	0.009988	0.002537	145	7
47.22	0.6	0.6	0.005	0.009543	0.002004	121	8

VI. ANALYSIS OF RESULTS

The results for the optimum topology of neural network is also in the form of special files saved using Easy NN-plus. Special files contain information about the number and values of weights, the input training patterns, and cycle by cycle learning process. In testing phase Easy NN-plus requires a text file containing the test samples. It also generates a text file, which contains the normalized outputs of test samples. In the result analysis the learning rate and momentum rate play an important role. Below figures shows the effect of changing of learning rate and momentum rate:

In the fig. 3 shows network 1with double hidden layers with 45.20 nodes and set the learning and momentum rate both 0.6 &0.6 respectively. The training stops when the network achieves the target error (i.e. 0.005). For this network the training is completed in 147 cycles.

When we change the momentum rate from 0.6 to 0.8 and fix the learning rate to 0.6, the network is able to achieve the target error 0.005 in approximate 341 cycles which is very much large. The effect of change of momentum rate is in shown in fig. 4. Similarly, very high value of the learning and momentum rate lead to the oscillations in the observations as show in fig. 5:

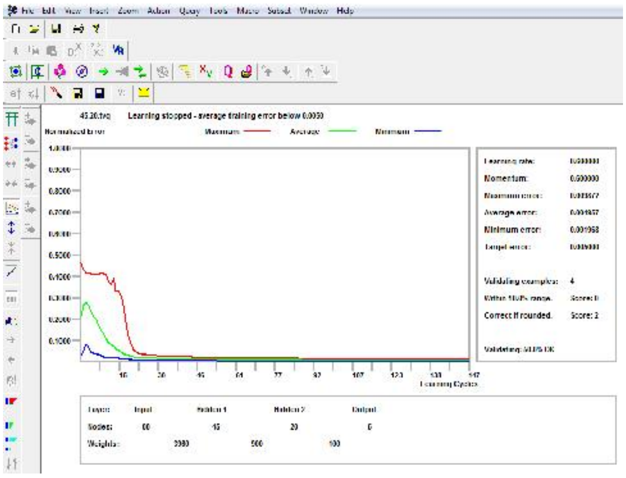

Fig. 3: learning graph for network 1 (Double layer with L.R. 0.6 & M.R. 0.6)

In this graph, the network uses learning rate of 1.0 and momentum rate of 0.9 respectively. These values are also not desirable, as the network in this case never trained. So the Choice of learning and momentum rate is an important design issues not only in achieving fast training, but also in obtaining good generalization. The optimum value of these parameters has to be achieved by hit and trail only as there is no specific rule for setting these values.

Although increment in the values of learning rate and momentum rate is supposed to decrease the time of network training, but this is not found to be always true after observing graph in fig. 6. In this fig. The effect of change of learning rate and momentum rate on double hidden layer network is shown.

For the same training data, but with different topology, the optimum value of learning and momentum rate is slightly different as shown in fig. 7. In this case three hidden layers are used with 60, 41 &22 nodes respectively. The optimum value of learning rate and momentum rate in this case are 0.6 & 0.6 respectively.

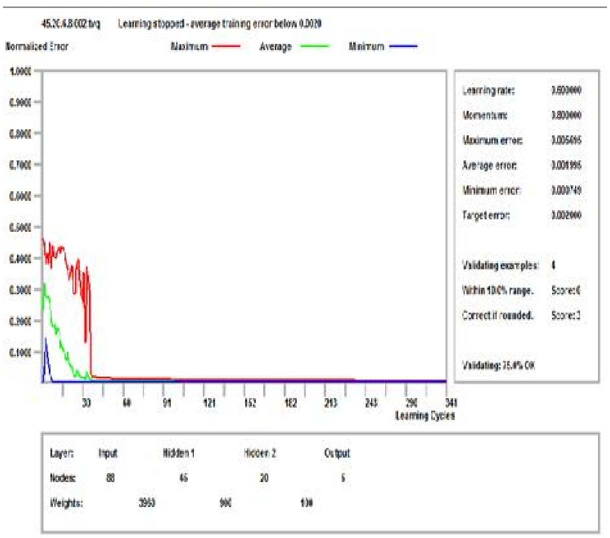

Fig. 4: learning graph for network 2 (Double layer with L.R. 0.6 & M.R. 0.8)

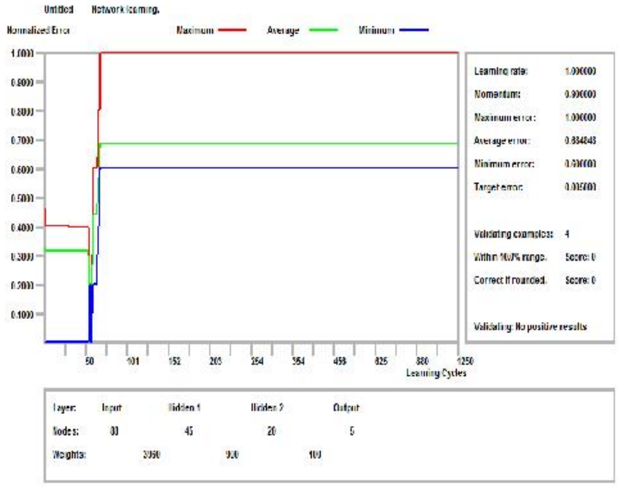

Fig. 5: learning graph for network 3 (Double layer with L.R. 1.0 & M.R. 0.9)

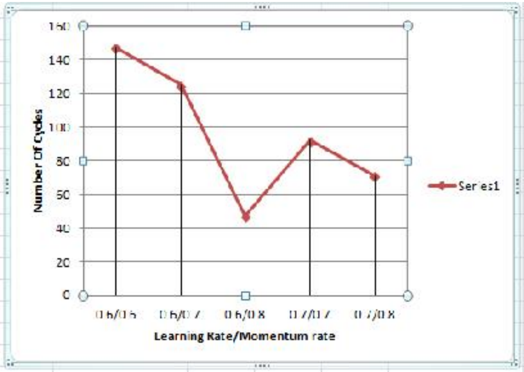

Fig. 6: Effect of learning & Momentum Rate on Double Hidden Layer Network

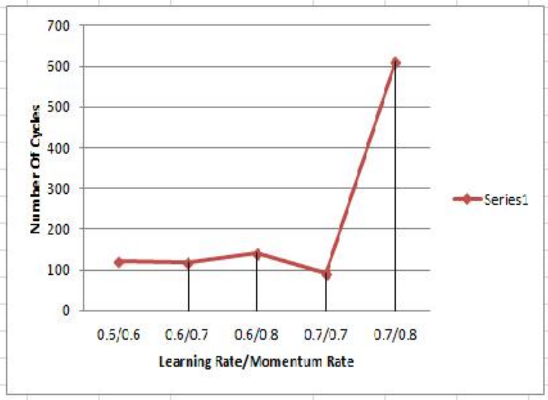

Fig. 7: Effect of learning & Momentum Rate on Triple Hidden Layer Network

VII. CONCLUSION & FUTURE WORK

The Offline signature is not an easy task. There are two main approaches for performing signature recognition. In this paper we purposed neural approach for offline signature recognition. In this research the result is provided by using 80 signatures. In which 70 signatures are used for training purpose and 10 for testing purpose. After doing fair amount of training and testing through the networks, the research for optimum ANN model for offline signature recognition concludes as, a network with triple hidden layers (i.e. 60.41.22) nodes, 100 no. of input nodes, 05 no. of output nodes, Learning Rate 0.6, Momentum Rate 0.6 and Target error for training is 0.005. It is conclude form above results that the neural network is able to recognize the offline signature with good accuracy.

Offline signature recognition is a difficult problem, but our hope is that the purposed system will be steps towards a neural network approach to robustly solve it. In future work the size of elements of rectangular matrix is decreased and in this way we get more values of 0's and 1's for a signature also we set the target error 0.002 instead of 0.005. In this way we increase the accuracy of network.

REFRENCES

[1] See.N.G," A Neural Network Approach for Offline Signature Verification" IEEE TENCON, pp. 770-773, 1993.

[2] Adel S; El-fishawy "scale-invariant adaptive lattice character recognition"(C4) ,Proceedings of the tenth national radio science conference FEB-16-18-1993.

[3] Richard-P. Lippmann (l989) "Pattern classification using neural network", IEEE communication Magazine,Nov 1996.

[4] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed. New York: Prentice-Hall, 1995.

[5] Rasha Abbas, B. Eng "*Backpropagation Networks Prototype for Offline Signature Verification*", Thesis report, RMIT, University of Melbourne, Australia, 1994.

[6] E. J. R. Justo, F. Bortolozzi and R. Sabourin, "*Off line Signature Verification using HMM for Random, Simple and Skilled orgeries*", ICDAR 2001, International Conference on Document Analysis and Recognition, vol. 1, pp. 105-110, 2001.

[7] Emre Ozgiundizz, Tulin, Senturk and M.Elif Karshgil, "*Off line Signature Verification and Recognition by Support Vector Machine*", European Signal Processing Conference, EUSIPCO 2005.

[8] H.Baltzakis, N.Papamarkos, "*A New Signature Verification Technique based on a Two-stage Neural Network Classifier*", Pergoman, pp. 95-103, 2001.

[9] Rohan K. Rajakakse, A. Ruvan Weeasinghale Kevin Senvirat, "*A Neural Network based Character Recognition for Sinhala Script*", SEARCC, 1996.

[10] Easy NN simulator (User's Manual)