An Approach to Recognize Characters based on Back Propagation Using Neural Network

[1]Neha Sharma, [2]Neeraj Dahiya
[1,2]South Point Institute of Technology and Management, India

Abstract
The main driving force behind neural network researchis the desire to create a machine that works similar to the mannerour own brain works. Neural networks have been used in avariety of different areas to solve a wide range of problems.Unlike human brains that can identify and memorize thecharacters like letters or digits; computers treat them as binarygraphics. Therefore, algorithms are necessary to identify andrecognize each character. In this article, we are going to look at amethod for handwritten character recognition was performed in order to improve the learning capability and accuracy of classification. Training data is first analyzed of line and information which differentiate them most is selected.Selected information determines characteristics of the neural networks.
Characters using back propagation algorithm. A Backpropagation network consists of at least three layers of units, aninput layer, at least one intermediate hidden layer, and an outputlayer. When a Back propagation network is cycled, an inputpattern is propagated forward to the output units through theintervening input-to-hidden and hidden-to-output weights.

Keywords
Neural Network, ANN, Neuron, Knowledge, Supervised, Pattern Recognition, Image Processing, Handwritten Character Recognition, Back Propagation Network

I. Introduction
A Neural network is a machine that is designed to model the way in which the brain performs a particular task or function of interest: The network is usually implemented by using electronic components or is simulated insoftware on a digital computer. " A neural network is a massively parallel distributed processor made up ofsimple processing units which has a natural propensity for storing experiential knowledge and making itavailable for use. It resembles the brain in two respects:

1. Knowledge is required by the network from its environment through a learning process.
2. Interneuron connection strengths, known as synaptic weights, are used to store the acquired knowledge".

II. Back Propagation Algorithm
Backpropagation, an abbreviation for "backward propagation of errors", is a common method of training artificial neural networks. From a desired output, the network learns from many inputs, similar to the way a child learns to identify a dog from examples of dogs.
It is a supervised learning method, and is a generalization of the delta rule. It requires a dataset of the desired output for many inputs, making up the training set. It is most useful for feed-forward networks (networks that have no feedback, or simply, that have no connections that loop). Backpropagation requires that the activation function used by the artificial neurons (or "nodes") be differentiable.The goal of any supervised learning algorithm is to find a function that best maps a set of inputs to its correct output. An example would be a simple classificationtask, where the input is an image of an animal, and the correct output would be the name of the animal. Some input and output patterns can be easily learned by single-layer neural networks (i.e. perceptrons). However, these single-layer perceptrons cannot learn some relatively simple patterns, such as those that are notlinearly separable. For example, a human may classify an image of an animal by recognizing certain features such as the number of limbs, the texture of the skin (whether it is furry, feathered, scaled, etc.), the size of the animal, and the list goes on. A single-layer neural network however, must learn a function that outputs a label solely using the intensity of the pixels in the image. There is no way for it to learn any abstract features of the input since it is limited to having only one layer. A multi-layered network overcomes this limitation as it can create internal representations and learn different features in each layer. The first layer may be responsible for learning the orientations of lines using the inputs from the individual pixels in the image. The second layer may combine the features learned in the first layer and learn to identify simple shapes such as circles. Each higher layer learns more and more abstract features such as those mentioned above that can be used to classify the image. Each layer finds patterns in the layer below it and it is this ability to create internal representations that are independent of outside input that gives multi-layered networks its power. The goal and motivation for developing the backpropagation algorithm is to find a way to train multi-layered neural networks such that it can learn the appropriate internal representations to allow it to learn any arbitrary mapping of input to output.
The backpropagation learning algorithm can be divided into two phases: propagation and weight update.

A. Phase 1: Propagation
Each propagation involves the following steps:

1. Forward propagation of a training pattern's input through the neural network in order to generate the propagation's output activations.
2. Backward propagation of the propagation's output activations through the neural network using the training pattern target in order to generate the deltas of all output and hidden neurons.

B. Phase 2: Weight Update
For each weight-synapse follow the following steps:

1. Multiply its output delta and input activation to get the gradient of the weight.
2. Subtract a ratio (percentage) of the gradient from the weight.

This ratio (percentage) influences the speed and quality of learning; it is called the learning rate. The greater the ratio, the faster the neuron trains; the lower the ratio, the more accurate the training is. The sign of the gradient of a weight indicates where the error is increasing, this is why the weight must be updated in the opposite direction.
Repeat phase 1 and 2 until the performance of the network is satisfactory.

Algorithm

Algorithm for a 3-layer network (only one hidden layer):

initialize network weights (often small random values)
do
forEach training example ex
prediction = neural-net-output(network, ex) // forward pass
actual = teacher-output(ex)
compute error (prediction - actual) at the output units
compute Δw_h for all weights from hidden layer to output layer // *backward pass*
compute Δw_i for all weights from input layer to hidden layer // *backward pass continued*
update network weights
until all examples classified correctly or another stopping criterion satisfied
return the network

III. Train Neural Network

When the network weights and biases are initialized, the network is ready for training. The multilayer feedforward network can be trained for function approximation (nonlinear regression) or pattern recognition. The training process requires a set of examples of proper network behavior—network inputs p and target outputs t.

The process of training a neural network involves tuning the values of the weights and biases of the network to optimize network performance, as defined by the network performance function net.performFcn

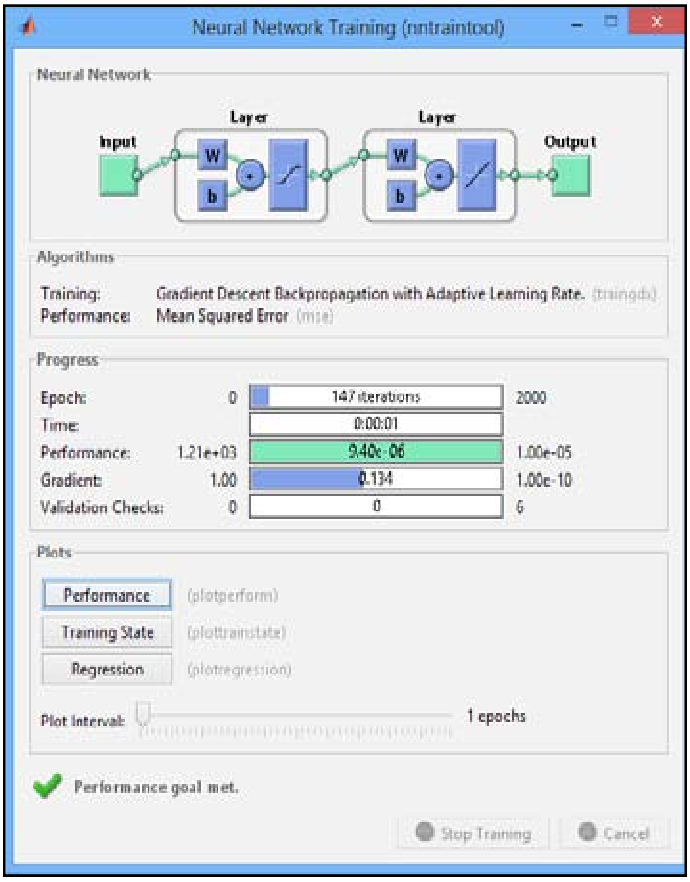

Fig. 1: Neural Network Training

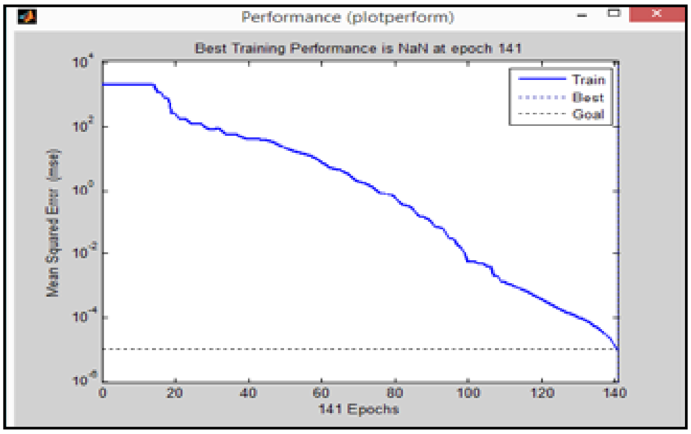

Fig. 2: Performance Graph

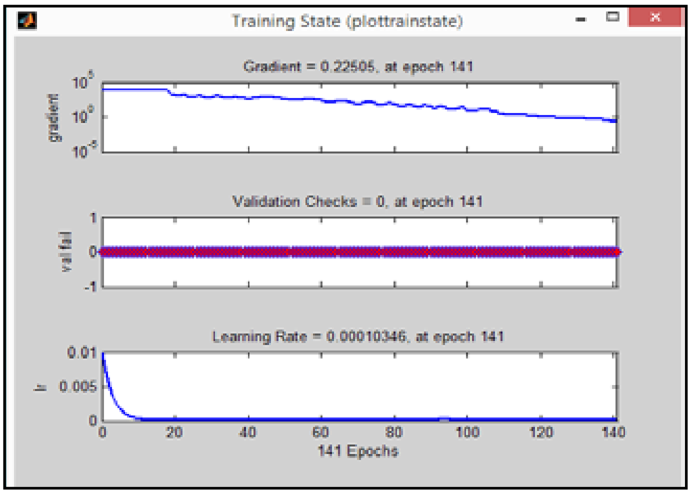

Fig. 3: Training State

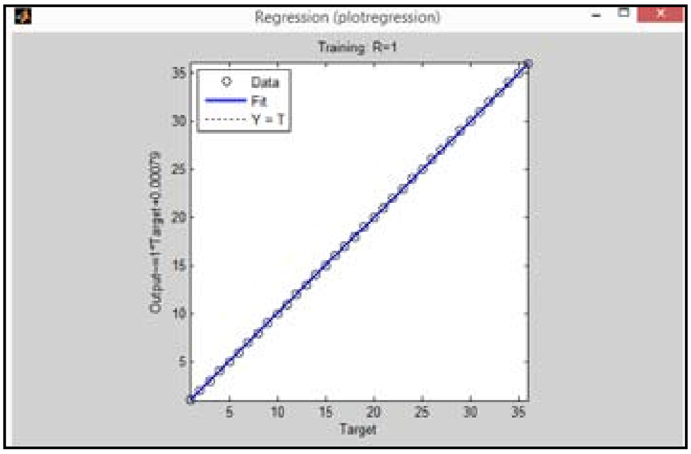

Fig. 4: Regression Plot

IV. Result Analysis

In this paper some printed and some handwritten character set and digits are taken for recognition. Algorithm steps are followed to achieve best accuracy and minimizing training time. We have trained the network through some samples.

Rest of the samples are used for testing the network. We have achieved best accuracy and minimized the training time. It is shown here with the help of graphs.Number of epochs taken to achieve the performance goal of 0.00001

In Performance graph we can see that we have achieved better accuracy. To find out the accuracy we have calculated the difference between actual result and desired result. Percentage accuracy is calculated using following formula:

$$\%Accuracy = \frac{Noofcharactersfoundcorrectly}{Totalnooftestingdata} * 100$$

The detail of experiment conducted is shown in the Table 1.

Table 1: Experimental Result Using Neural Network

Parameters	Experiment Result
Convergence Objective	0.0001
Learning Rate	0.01
Training method used	Trainlm(Levenberg-Marquardt)
No of training data	36
No of testing data	20
No of epoch taken to converge	3
No of characters found correctly	19
Accuracy	97%

A. Test Image

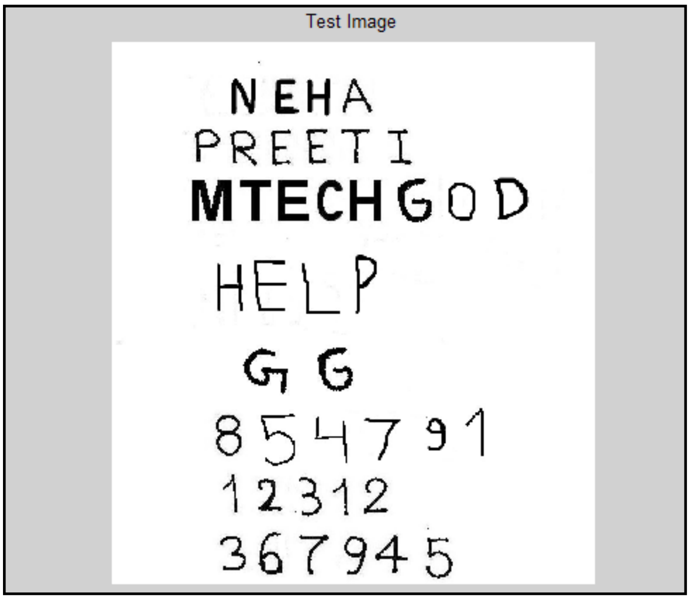

Fig. 5: Input Data

B. Recognized Data

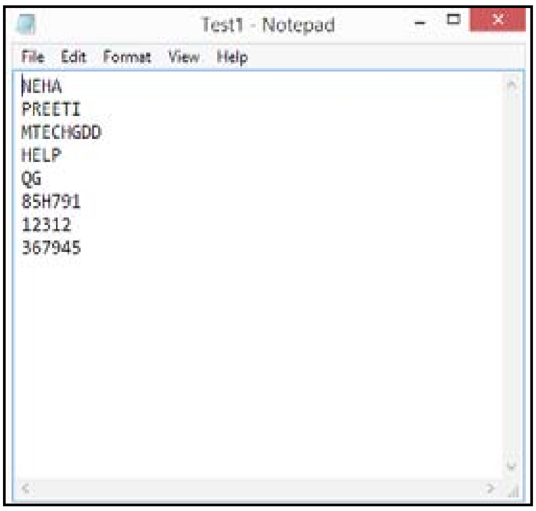

Fig. 6: Output Data

V. Conclusion and Future Work

Neural Network has a wide application in the area of pattern recognition. In this paper, a system for recognizing printed English alphabets and digits has been developed. A good accuracy of 97% has been achieved with the help of back propagation algorithm. Network must be trained with ample data so that good recognition accuracy can be achieved.

This algorithm can be used for recognition of handwritten hindi, English, Arabic, tamil characters. In future, more image processing techniques can be used to improve our result. Different feature extraction methods can be used to improve the recognition accuracy.

References

[1] S.P. Kosbatwar, S.K Pathan,"Pattern Association for Character Recognition by Back Propagation algorithm using Neural Network Approach", International Journal of Computer Science & Engineering survey, Vol. 3, No. 1, February 2012.

[2] Vinita Dutt, Sunil Dutt,"Hand Written Character Recognition Using Artificial Neural Network", Journal of Advances in Computing, Vol. 1, No. 1, pp. 18-23, 2011.

[3] G. G. Rajput, S. M. Mali,"Marathi Handwritten Numeral Recognition using Fourier Descriptors and Normalized Chain Code", International Journal of Computer Application(IJCA) on Recent Trends in Image Processing and Pattern Recognition (RTIPPR), pp. 141-145, 2010.

[4] C. Suresh Kumar, Dr. T. Ravichandran,"Handwritten Tamil Character Recognition Using RCS Algorithm", International Journal of Computer Applications, Vol. 8, No.8, October 2010.

[5] Youfu Wu, Yongwu Wu, Gang Zhou, Jing Wu, "Recognizing Characters Based on Gaussian-Hermite Moments and BP Neural Networks", IEEE International Conference on Intelligent Computation Technology and Automation (ICICTA), Vol. 2, pp. 992-995, 2010.

[6] G. G. Rajput, RajeswariHorakeri, Sidramappa Chandrakant,"Printed and Handwritten Kannada Numeral Recognition Using Crack Codes and Fourier Descriptors Plate", International Journal of Computer Application (IJCA) on Recent Trends in Image Processing and Pattern Recognition (RTIPPR), pp. 53-58, 2010.

[7] MatLab R2010 Neural Network Tool Box Product Help

[8] Rashad Al-Jawfi,"Handwriting Arabic Character Recognition LeNet Using Neural Network", International Arab Journal of Information Technology, Vol. 6, No. 3, July 2009.

[9] Srinivasa Kumar Devi Reddy, SettipalliAppaRai,"Hand written character recognition using back propagation network", Journal of Theoretical and Applied Information Technology, 2009.

[10] S K Hasnain, Muhammad SamiullahAwan,"Recognizing Spoken Urdu Numbers Using Fourier Descriptor and Neural Networks with Matlab", IEEE Second International Conference on Electrical Engineering, 25-26 March 2008

[11] Velappa Ganapathy, Kok Leong Liew,"Handwritten Character Recognition Using Multiscale Neural Network Training Technique", World Academy of Science, Engineering and Technology 15 , pp. 32-37, 2008.

[12] Feng Yang, Fan Yang,"Character Recognition Using Parallel BP Neural Network", IEEE International Conference on Audio, Language and Image Processing, pp. 1595-1599, 2008.

[13] J. Sutha, N. Ramaraj,"Neural Network Based Offline Tamil Handwritten Character Recognition System", IEEE International Conference on Computational Intelligence and Multimedia Applications, Vol. 2, pp. 446-450, 2007.
[14] Cheng-Lin Liu,"Normalization-Cooperated Gradient Feature Extraction for Handwritten Character Recognition", IEEE Transaction on pattern analysis and machine intelligence, Vol. 29, No. 8, Aug. 2007.