ISSN:2229-6093

RECOGNIZING HANDWRITTEN NUMERALS USING MULTILAYER FEED FORWARD BACKPROPAGATION NEURAL NETWORK

P.PANDI SELVI,
RESEARCH SCHOLAR,
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING,
ALAGAPPA UNIVERSITY
KARAIKUDI,TAMILNADU,INDIA.
selvikrish.selvi@gmail.com

Dr.T.MEYYAPPAN,
PROFESSOR,
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING,
ALAGAPPA UNIVERSITY
KARAIKUDI,TAMILNADU,INDIA.
meyslotus@yahoo.com

Abstract

Image processing is simply the processing of the given image. The input is just an image, that may be from any source, and the output may be an image or a set of parameters that are related to that particular image. Recognition plays an important role in the area of image processing. In this research work, the authors focus on the recognition of handwritten digits. A new method that uses neural network is proposed for recognizing handwritten numerals. It consists of three phases namely preprocessing, training and recognition. Preprocessing stage performs noise removal, binarization, labeling, rescaling and segmentation operations. Training stage adopts backpropagation with feedforward technique. Recognition stage recognizes input images of numerals. The proposed method is implemented in Matlab. The method recognizes the numerals with accuracy in the range 95-100%. It performs well and maintain the accuracy level even in case of deformed images and images of any size.

Keywords

Multilayer feed-forward back propagation neural network, Gaussian filter, OTSU's method, Bilinear interpolation method, Floyd-Warshall's algorithm.

1. Introduction

Image processing is classified into two broad areas as, computer graphics and computer vision. In computer graphics, the inputs are usually manmade from various objects and lighting. Whereas in computer vision, the inputs are usually extracted from videos, a computer, camera or a software. It includes various methods for acquiring, processing and analyzing images. The various tasks it includes are, recognition, motion analysis, scene reconstruction and image restoration, and in the case of recognition it includes object recognition, identification and detection. In this research work, a new neural network method is proposed to recognize handwritten numerals. Scanned images of handwritten numerals are input for recognition. The recognition operation comprises 3 stages: i) Preprocessing ii) Training iii) Recognition. Training and recognition employs feedforward backpropagation algorithm. Fengwei An et.al[8] provided hardware solution with FPGA. The new method proposed in this research work implements the recognition with software coding. It adopts Multilayer feedforward neural network architecture. The reason for using feed-forward neural network is that, they are the simplest form of associative memory, and they have the capability to learn from experience, generalize from previous examples to new ones and also extract essential characters or features from various inputs that contain irrelevant data. The proposed method is applied over different input images. The recognition accuracy is consistent even for deformed images.

2. Related Work

In 1990, Y.Le Cun, et.al[1], proposed a Back-propagation network for recognizing handwritten digits. The preprocessing stage here includes Acquisition of the data, binarization, location of the zipcode, preliminary segmentation and finally the normalization of the digits using linear transformation method. The recognition task is performed by the multi-layer network, as well as the connections within the network are adaptive and the network was trained with back-propagation network. With their approach the error rate of the training set was 3.4% and the MSE was 0.024. The results show that the method can be extended for larger applications.

In 1998, Daniel Cruces Alvarez, et.al[4], proposed a neural network method for recognizing printed and handwritten digits. The recognition is performed using a multilayer and clustered back propagation algorithm. For extracting the features kirsch masks are adopted and for classifying the numerals, they used five independent subnetworks. The main purpose of using multilayer neural network is to minimize the mean square error, between the arrived output and the desired output. Here each subnet between the input and the hidden layer are initialized with random weights and also trained with different feature maps. All the connections here are adaptive in nature and are generally trained using backpropagation algorithm.
Here they provided another neural classifier, termed the refinement network, which acts mainly on the patterns that are rejected by the first network, and that is done mainly to improve the output more better. The refinement network here is made up of 45 neurons. Each one is trained to distinguish between the numerals. Thus with their method, the rejection rate is 9% and the error rate is reduced to 1%.

In 2009, Calin Enachescu and Cristian- Dumitru Miron[6], proposed a neural computing method for recognizing handwritten digits. A framework was presented by them to classify handwritten digits, and the classification was performed using Convolutional Neural network. It was mainly designed to recognize patterns from pixel images directly with minimal preprocessing. CNN is a feed forward neural network. Features are extracted from the input image, in the first hidden layer and the patterns are classified in the final hidden layer. The number of hidden layer neurons can be varied to control the learning capacity as well as the generalization capacity of the classifier. With CNN it is possible to extract information from images with minimal preprocessing. Apart from this, it is also necessary to remove unwanted pixels from the background, which improves the performance not only during the learning process, but also when using CNN in normal. In this case they encoded the image, that are used in the learning process, in which the white pixels are set to 1 and the background pixels are set to 0. The parameter which is affecting the learning process is the learning rate. The higher the learning rate, learning performance is low, whereas with small learning rate, the learning performance is better. After the learning process is over, they tested the network with different images. With original NIST dataset, it provided 96.74% accuracy, and with the images that are without background it provided 96.56% accuracy.

In 2010, Dewi Nasien, et.al[7], proposed a method for recognizing handwritten latin characters using Freeman chain code Representation and Feedforward Neural network classifier. A randomized algorithm has been generated for characters with the problem of chain code, which mainly minimizes the length of the chain code. It generally undergoes three stages as pre-processing, feature extraction and classification. During the preprocessing stage, they used thinning method for obtaining the skeleton of a character, which also removes any redundant information, as well as it maintains the features of the image. For feature extraction, they proposed a randomized algorithm, to eliminate any redundancy, that is to minimize the length of the chain code, with the data. And finally for classification they used a neural network classifier.

In 2011, Fengwei An, et.al[8], proposed an FPGA implemented associative-memory based online learning method for character recognition. As a first step it undergoes preprocessing stage, that includes the following stages. As the characters are from the scanned images, there are possibilities to affect the character's clarity. To remove such noise, they used 3x3 Gaussian filter. For binarizing the given color image OTSU's method was used, in which the given image was classified into foreground and background and also an optimum threshold was set to separate these two classes. After binarization, it undergoes labeling stage and is performed with Floyd-Warshall's algorithm and according to the label number, they are then segmented. These images were different from each other in size and position. They are then rescaled to 16x16 pixels with bilinear interpolation method.

The extracted features are then used to calculate the Manhattan distance between the inputs and the reference patterns. Here, they had chosen direction gradient feature, and is calculated by convolving two 3x3 Sobel operators. For calculating Manhattan and Hamming distance, they introduced a fully parallel mixed digital/analog architecture for calculating the nearest distance. The distance are mapped to time-domain concept using adjustable ring oscillators. The associative memory with this time domain Winner-Take-All circuit, detects the first occurred ring oscillator signal as the winner and is calculated accordingly. For finding Euclidean distance they implemented associative memory in an FPGA which is a programmable VLSI device. For online learning, they used short/long term memory concept, in which with this they were able to eliminate irrelevant attributes, as well as the requirement for storing large amount of data is also minimized through a forgetting process in short-term memory.

The online learning model operates in the following way. First, the normalized binary image is taken as input. Euclidean distance is then calculated with an FPGA implemented associative memory, for classifying the relevant data. Next the learning starts, in which the winner distance is calculated first and if it is equal to the threshold ie., if the input matches the reference pattern in memory, it gets transferred to a predefined location. And if it does not matches the pattern, it is transferred to the reference memory and the ranking algorithm transfer it to the top of long term or short term memory. With their approach the mismatch rate was reduced to 0.6%.

In the proposed work, the above method is implemented with Matlab coding. The proposed method adopts backpropagation with feed forward technique.

3. Proposed Research Work

The three stages in the work are (i) Preprocessing, (ii) Training and (iii) Recognition. Steps in the proposed method is detailed in the following sections.

3.1. Preprocessing Stage

Step 1: Any noise in the input image is removed with the help of filters. A 3X3 Gaussian filter is used, where it removes noise both horizontally and vertically. The final result is the product of two Gaussians along each direction, and is calculated by,

$$g(x,y) = \frac{1}{2\pi\sigma^2} . e^{(-x2 + y2) / 2\sigma2} \qquad (1)$$

where x is the distance from the origin in the horizontal axis, y is the distance from the origin in the vertical axis, and σ is the standard deviation.

The standard deviation is calculated using,

$$\sigma = \sqrt{\frac{\Sigma^{n}_{i-1}}{n-1} - \frac{n}{n-1}\left((\Sigma^{n}_{i=1} a_i)/ n \right)^2} \qquad (2)$$

Step 2: The filtered color images are then converted into a binary image by OTSU's method. It is calculated as follows

$$\sigma^2_b(t)=\sigma^2 - \sigma^2_\omega(t)=\omega_1(t)\omega_2(t)[\mu_1(t)-\mu_2(t)] \qquad (3)$$

Where ω_i is the class probability and μ_i the class means. Initially the values of ω_i and μ_i are set to zero. Next it steps up through the threshold values of t =1,2, …, n and the values of ω_i and μ_i are updated periodically and the value of σ^2_b (t) is calculated. σ^2_b (t) is the desired maximum threshold. The class probability ω_1(t) is calculated as

$$\omega_1(t) = \Sigma^t_0 p(i)$$

and the class means by

$$\mu_i(t) = \Sigma^t_0 p(i)\, x(i)$$

As a next step two maximums are computed as the greater maximum σ^2_{b1} (t) and the greater or equal maximum σ^2_{b2} (t). The desired threshold is then calculated by finding the mean of these two values.

Step 3: A label number is then provided for each binary-connected components, and those labels that share similar characteristics are segmented.

The shortest path between each binary component is then find by,

Shortest path(i,j,k)=min(shortestpath(i,j,k-1), shortestpath(i,k,k-1)+shortestPath(k,j,k-1)). **(4)**

It first finds the shortestPath(i, j, k) for all (i, j) pairs for k = 1, and then k = 2, etc. This continues until *k* = *n*.

Step 4: These binary images are rescaled by bilinear interpolation method and then extracted. To find the value of the unknown function f at the point P = (x, y). It is assumed that the value of f at the four points $Q_{11}=(x_1, y_1)$, $Q_{12}=(x_1, y_2)$, $Q_{21}=(x_2, y_1)$, and $Q_{22}=(x_2, y_2)$. First linear interpolation is applied in the x-direction by,

$$f(R_1) \approx \frac{x_2-x}{x_2-x_1} f(Q_{11}) + \frac{x-x_1}{x_2-x_1} f(Q_{21}) \quad (5)$$

Where $R_1=(x,y_1)$

$$f(R_2) \approx \frac{x_2-x}{x_2-x_1} f(Q_{11}) + \frac{x-x_1}{x_2-x_1} f(Q_{21}) \quad (6)$$

Where $R_2=(x,y_2)$

Interpolation in y-direction is found by,

$$f(P) \approx \frac{y_2-y}{y_2-y_1} f(R_1) + \frac{y-y_1}{y_2-y_1} f(R_2) \quad (7)$$

After calculating the interpolation value in the x and y direction, the desired interpolation value f(x,y) is found by using equations 5,6 and 7 as,

$$f(x,y) \approx \frac{f(Q_{11})(x_2-x)(y_2-y)}{(x_2-x_1)(y_2-y_1)} + \frac{f(Q_{21})(x-x_1)(y_2-y)}{(x_2-x_1)(y_2-y_1)} + \frac{f(Q_{12})(x_2-x)(y-y_1)}{(x_2-x_1)(y_2-y_1)} + \frac{f(Q_{22})(x-x_1)(y-y_1)}{(x_2-x_1)(y_2-y_1)} \quad (8)$$

Step 5: Finally, the extracted binary images are fed into a multilayer feed-forward backpropagation neural network for recognizing digits.

3.2 Training the Network

The simplest among the associative memory model is the feed-forward type of neural network. A multi-layer feed-forward back propagation neural network is used for recognizing handwritten numerals. The training network consists of three layers the input layer, hidden layer and the output layer. An input vector X is applied to the network through the input layer and the output Y is produced across the output layer. In the case of multilayer network, the **NET** value is calculated for each neuron layer by layer using,

$$NET=XW \quad (9)$$

where X is the input and W is the weight value.

For processing the NET signal to further layers, an activation function is applied to the NET value, and the OUT value is calculated by,

$$OUT=F(XW). \quad (10)$$

Here for example the target value is set to 1. If the calculated OUT value is not greater than or equal to 1, the desired target value, the backpropagation algorithm is applied and the associated weights are adjusted using the Delta rule.
The Delta value is calculated using the formula,

$$\delta=OUT(1-OUT)(Target-OUT) \quad (11)$$

Where (Target-OUT) gives the error signal, it is then multiplied by the squashing function OUT(1-OUT). After the Delta value is calculated, it is multiplied with the OUT value of the desired layers neuron, which is further multiplied with the training rate coefficient η as,

$$\Delta W=\eta.\delta.OUT \quad (12)$$

The result thus obtained is added to the weight value of the corresponding neuron in the hidden layer to the output layer by,

$$W(n+1)=W(n)+\Delta W \quad (13)$$

Where W(n) is the value of weight from neuron in the hidden layer to the neuron in the output layer, and W(n+1) is the weight value after adjustment. After the weight value is adjusted, it is multiplied with the input vector X in the hidden layer and the OUT value is calculated. The network structure is as shown in Fig 1.

ISSN:2229-6093

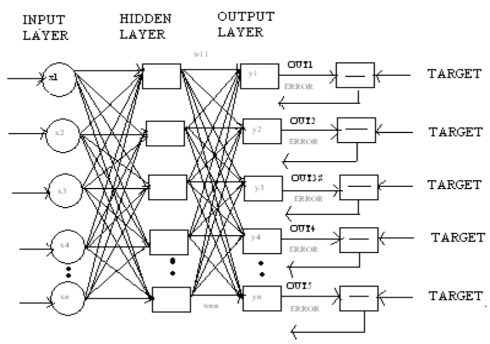

Fig 1: The Network structure.

The training process continues until the error value and the loss function are reduced to a minimum, and it occurs at the eleventh epoch, where the error value is reduced to 0.09095 and the loss function is reduced to 0.076, and it is shown in the following Fig 2.

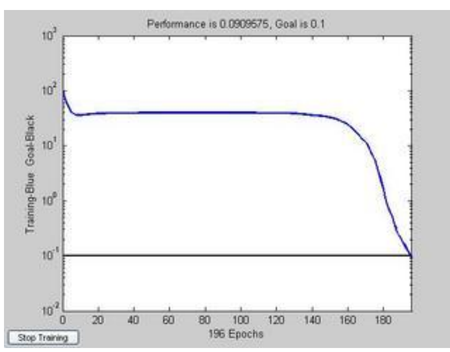

Fig 2. Training process

3.3. Recognition of the Numerals

Once the training is over, the preprocessed images are then fed into the network for recognition. If the given input matches with the trained data stored in memory, the appropriate image of the numeral is extracted and displayed. In certain situations , if the given input is deformed or distorted, and also the calculated OUT value not the desired one, the backpropagation algorithm is then applied, such that it propagates backwards and the weight value from the hidden layer to the output layer is adjusted and the OUT value is recalculated.

4. Implementation And Results

The proposed method is tested with the following set of images shown in Fig 3. Any one numeral is chosen from the set and subjected to the proposed method.

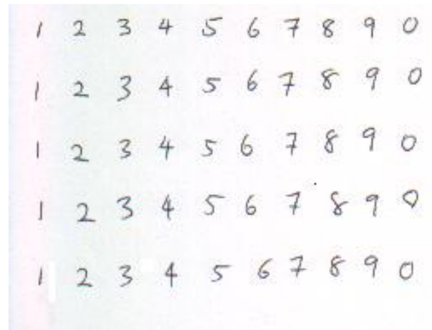

Fig 3: Inputs for recognition.

Resulting image after applying Gaussian filter for removing any noise, is shown in Fig 4. The filtered color image is then converted to a binary image with OTSU's method. The converted binary image is shown in Fig 5.

Fig 4. RGB digits that are selected for recognition.

Fig 5. Binary digits.

The binary images are then provided with a label number by Floyd-Warshall's algorithm and are segmented according to the similarity in the label number. These segmented images are then rescaled by bilinear interpolation method and the images are extracted. The extracted image is shown in Fig 6.

Fig 6. Extracted digits.

The extracted images are then fed into the neural network architecture for recognition. We used different kinds of inputs with various shapes. Even then we were able to recognize the digits exactly with better accuracy. The recognized digits are shown in Fig7.

1 2 3

Fig 7. Recognized digits.

5. Conclusion

In this paper, the authors implemented a multi-layer feed-forward backpropagation network that recognizes handwritten numerals. Given input image of numerals undergoes three stages namely preprocessing, training and recognition. Preprocessing stage plays a vital role and influences the accuracy of recognition. Images of various sizes, shapes undergo preprocessing stage first. Then, the multi-layered feed-forward back propagation network is trained with preprocessed images. The final recognition stage recognizes images of any size and shape as input to the method. The proposed method is implemented with matlab coding. Digitized handwritten images are tested with the proposed method and the recognition is found to be 95-100% accurate. Similar strokes of two numerals are recognized as one for the other, was the drawback of this method. Further research is in progress to eradicate this negative recognition.

References

1.) Y.Le Cuu, B.Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard and L.D. Jackel," Handwritten Digit Recognition with a Back-Propagation Network," AT&T Bell Laboratories, Holmdel, N.J.07733 1990.

2.) M.D.Garris, R.A. Wilkinson and C.L.Wilson," Methods for Enhancing Neural Network Handwritten Character Recognition," International Joint Conference on Neural Networks, Volume I, IEEE, Seattle, July 1991.

3.) S.Knerr, L. Personnaz, G.Dreyfus," Handwritten Digit Recognition by Neural Networks with Single-Layer Training," IEEE Transactions on Neural Networks, vol.3, 962(1992).

4.) Daniel Cruces Alvarez, Fernando Martin Rodriguez, Xulio Fernandez Hermida,"Printed and Handwritten Digits Recognition Using Neural Networks", E.T.S.I.T. Ciudad Universitaria S/N.36200 Vigo.SPAIN, 1998.

5.) A.Akoum, B.Daya, P.Chauvet," Two Neural Network For License Number Plates Recognition," Journal of Theoretical and Applied Information Technology, 2005-2009 JATIT.

6.) Calin Enachescu, Cristian-Dumitru Miron,"Handwritten Digits Recognition Using Neural Computing", Scientific Bulletin of the Petru Maior University of Tirgu Mures, Vol.6(XXIII), 2009, ISSN 1841-9267.

7.) Dewi Nasien, Siti S. Yuhaniz, Habibollah Haron," Recognition of Isolated Handwritten Latin Characters Using One Continuous Route of Freeman Chain Code Representation and Feedforward Neural Network Classifier", World Academy of Science, Engineering and Technology 67 2010.

8.) Fengwei An, Hans Jurgen Mattausch, Tetsushi Koide," An FPGA-implemented Associative memory Based Online Learning Method," International Journal of Machine Learning and Computing, Vol 1, No.1, April 2011.

9.) S.M.Krishna Ganesh, P.Saranya,"An Evolutionary Neural Network Architecture Optimization Algorithm for Handwritten Digits Recognition", IOSR Journal of Engineering, Apr.2012, Vol.2(4) pp:726-730.

10.) J.V.S. Srinivas, P.Premchand," Handwritten Digit Recognition Using Elman Neural Network", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 5, May 2012.