



Hand Writing Numbers detection using Artificial Neural Networks

Ahmad Saeed Mohammad¹

Dr. Ahmed Khalaf Hamoudi²

Yasmin Abdul Ghani Abdul Kareem¹

¹ Computer & Software Eng., College of Engineering , Al- Mustansiriya Univ., Baghdad, Iraq

² Control & System Engineering, University of Technology, Baghdad, Iraq

Ahmad79sm@yahoo.com, ahmed_khh22@yahoo.com, yasa_alyozbaki@yahoo.com

Received: 31/10/2012

Accepted: 1/4/2013

Abstract – A fast, simple and effective method to recognize different hand writing numbers is presented. Hand writing recognition took high attention in the recent years by researcher of the intelligent systems, since it can be used in many applications such as car plate recognition and bank account checking. The purpose of this paper is to develop a method for hand writing numbers detection by using artificial neural network. The suggested work is divided mainly into four stages and is proposed to resolve the digits number (i.e., hand writing numbers). Image sample of hand writing numbers is acquired by a digital camera or scanner, and then it is converted by using the suggested work which is consisted of four stages to resolve digits. Artificial neural network (ANN) was applied to recognize the hand writing numbers. Learning method of the ANN is back propagation and all process handled by MATLAB language.

Keywords – Image processing, Artificial Neural Network, Back-propagation, Car plate recognition.

1. INTRODUCTION

Decimal number recognition is a challenging problem to researchers who work in this area in recent years.

There are many fields related to handwriting numbers detection, for examples, checks in banks and recognizing numbers in car plates. In other words, it helps the computer to understand the numbers that are written manually by users and to view them according to the computer processes.

Many methods had been proposed in research work of Hand Writing Numbers detection.

There is a batch of literature in this area. Some of the related work is as follows: [1] have developed a license plate recognition system. Their methods first separate each digit from the license plate using image processing tools, then built a classifier using a training set based on digit extracted from approximately 350 license plates. [2] Has built a semi-automatic name plate recognition system by using artificial neural network. Thin window scanning method was proposed to resolve the alphanumeric character. [3] Has proposed a method for heart extracting character without prior knowledge of their position and size. [4], [5] and [6] proposed some methods in order to recognize the plate's characters. In [7] the Conjugate gradient descent back-propagation algorithm (CGD-BP) is used for training purpose. CGD-BP differs from primary back-propagation algorithm in the sense that conjugate algorithms perform line search along different directions which produce faster convergence than primary back-propagation. Percentage Recognition Accuracy (PRA) and Mean Square Error (MSE) have been taken to estimate the efficiency of neural network to recognize the numerals. [8] has Proposed approach for English sentence recognition (ESR)

system. This approach focuses on the automatic system that recognizes continuous English sentence through a mouse-based gestures in real-time based on Artificial Neural Network. The proposed Artificial Neural Network is trained using the traditional back propagation algorithm for self supervised neural network which provides the system with great learning ability and thus has proven highly successful in training for feed-forward Artificial Neural Network.

2. SUGGESTED WORK

General Steps for Suggested Work:

The suggested work is divided into four stages, image acquisition, image preparing, ANN system and finally test the designed system and output the results.

2.1. Image Acquisition

All the digits were captured by a camera or a scanner in one image as shown in Figure (1). Segmentation and separate each digit from others. This process result in 100 separated digit. Each digit number has ten hand writing image of different thickness and shapes. Therefore for ten digit numbers from '0' to '9', one hundred cases had resulted to the next stage.

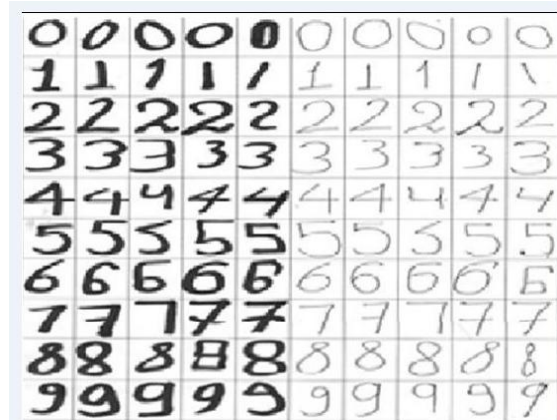


Figure (1): The hand writing numbers in one image as was captured by camera or scanner

2.2. Image Preparing

The image preparing process is divided into four stages: convert the size of image of a digit which comes from a camera or a scanner (after separated each digit from other digits) to a matrix of dimension (9 x 7), convert the matrix to a gray matrix, convert the type of matrix from gray to binary code, convert the binary matrix to a one binary vector of 63 elements (because $9 \times 7 = 63$). The above four stages repeat with all other digits, and then save the resulted binary vectors and the corresponding targets in a Mat file. Figure (2) shows two hundred cases, one hundred for gray image and another for it's conversion to binary images. Figure (3) shows the shape of the converted matrix image (left) to the Gray (right). Figure (4) shows the shape of the converted Gray (left) to the Binary (right).

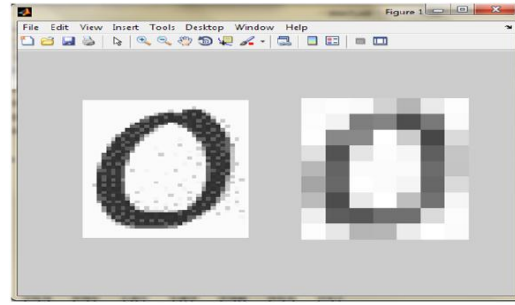


Figure (3): Converted matrix image (left) to binary (right)

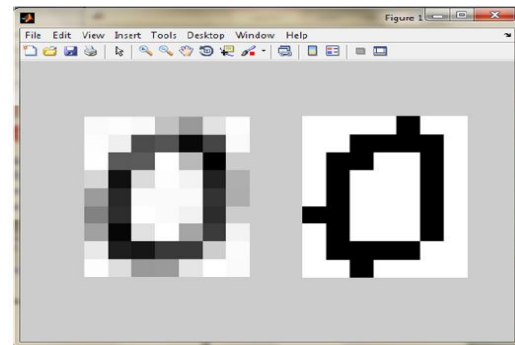


Figure (4): Converted binary (left) to gray matrix (right)

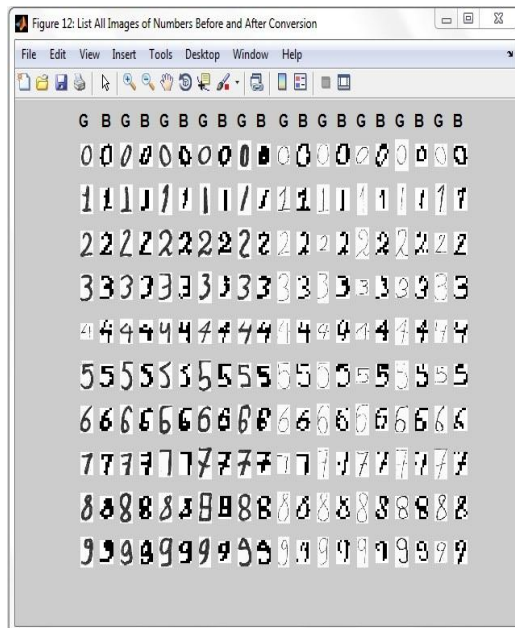


Figure (2): Representing of the Gray image (G) and its converted to Binary image (B)

2.3. Artificial Neural Networks (ANN)

The artificial neural network (ANN) was arranged into two stages; these are ANN design and ANN training.

2.3.1. ANN design

Artificial Neural Network has been created to have the following layers:

1-Input layer which has 63 neurons, because the input from the **Image Preparing** stage above is a vector of $9 \times 7 = 63$ elements for each digit number written by hand. Because we have 100 hand writing numbers, so we have 100 vectors (i.e., a vector for each hand writing number digit).

2- The Artificial Neural Network has two hidden layers because of the large number of inputs (100 vectors; each with 63 elements), the following numbers of neurons in the two hidden layers are suggested:

- a) First hidden layer has 15 neurons.
b) Second hidden layer has 10 neurons.
3- Output layer has only one neuron because the target was set to one element.
4- The Target is represented by an array of (1 x 100). The first 10 elements in the array are the target for the number '0'. We used only one element from the first 10 elements of the target array with each hand writing input digit number (i.e., with each one hand writing elements from 10 different in thickness and shapes) of the number '0' (see figure (1)). The second 10 elements in the target array are the targets for the number '1' and so on for other elements in the target.

The ANN was viewed in the MATLAB program, as shown in figure (5). The Artificial Neural Network had been trained according to batch training (Gradient Descent Back Propagation with Adaptive Learning Rate) as shown in table (1). The advantage of the batch training method is the ability of training all the 100 input vectors at the same time.

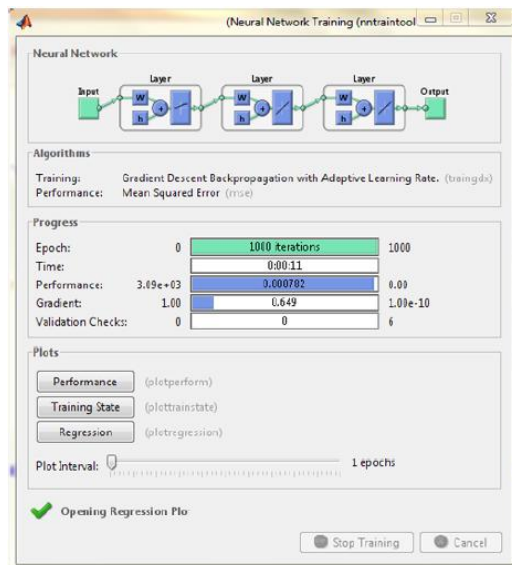


Figure (5): Training the program of ANN.

Table (1): ANN Specifications

Layer	No. of Nodes	Transfer Function
Input	63	-----
Hidden 1	15	Logistic Sigmoid
Hidden 2	10	Pure Line
Output	1	Pure Line
Performance Function		Mean Square Error
Training Function		Gradient Descent Back Propagation with Adaptive Learning Rate
Maximum Epochs		1000
Goal		1×10^{-10}

2.3.2. ANN Training

The artificial neural networks had been trained to get weights and biases, which are saved to avoid re- training it. These weights and biases will be used for testing the hand writing numbers in the next stage to classify the input as shown in figure (5).

2.4. Testing and Output

Before test, the designed system was made to record sound for each number in Arabic and English using a microphone. After training the neural network in above stage, the values of weights and biases were saved. These saved values of weights and biases are fixed (i.e., not changed during testing the input samples), and will be used to detect any hand writing number. The above four Steps for the Suggested Work will be applied when detect the hand writing numbers.

The ANN with its fixed weights and biases, and its input (the hand writing number sample) will be trained. If the output of the trained ANN matches one of the target cases, show the original image for this case (digit), and play the sound of this digit in Arabic and English. If the output does not match any one of the target cases, print message "Not Defined Digit".

The SUGGESTED WORK Algorithm

- 1: By using camera or scanner, capture all the digits of hand writing numbers in one image as shown in figure (1).
- 2: For the image presented in figure (1), make segmentation to separate each image of digit from others. This process results in 100 separated images which represent 100 hand writing digits.
- 3: Convert each image of digit from the above step to a (9x7) matrix.
- 4: Convert the above matrix to the (9x7) gray code matrix.
- 5: Convert the gray matrix to the binary code matrix of the same dimension.
- 6: Convert the binary matrix to a binary vector of 63 elements.
- 7: Repeat the steps from 3 to 6 with each digit resulted from step 2. This process results in 100 binary vectors of 63 elements.
- 8: Save the converted images (i.e., the binary vectors) in Mat file that represents the inputs to ANN.
- 9: Prepare the (1x100) elements of row vector target; one element for each shape of the hand writing numbers, and save it in a Mat file.
- 10: Make record of the human sounds of the numbers '0' to '9' in Arabic and English.
- 11: Save the recorded sounds (as vectors) in Mat file that will be used in ANN testing step.
- 12: Create a Neural Networks as specified by table 1.
- 13: Set the initial weights and biases that will be saved into Mat file.
- 14: Load (Inputs, Targets, Initial Weights and Biases).
- 15: Train initial weights and biases presented in figure (5) for the purpose of having a new trained weights and biases corresponding to the hand writing numbers (which are the inputs to ANN).
- 16: Save the above new trained weights and biases.
- 17: For test, load trained weight, trained biases, sounds of numbers, and any Samples Images of Hand Writing Numbers.
- 18: Simulate the ANN again using figure (5).
- 19: If the simulation output of the ANN matches one of the target cases, show the original image for this case.
- 20: Play the sound for this matched digit in Arabic and English.
- 21: If the output does not match one of the target cases, print message 'Not defined digit'.

3. EXPERIMENT AND RESULTS

Testing the image detection

Weights, biases, input binary vectors of hand writing numbers, and sounds were loaded into the simulated program to test the image detection, as shown in figure (5). This test result is presented in figure (6). This shows how the above suggested work can give high precision in detecting the hand writing number of digit '0'. In figure (6) the right image is the hand writing of the digit '0' that was used as input to the ANN. The left image is the original for the digit '0' that will be displayed after the detection of the hand writing of digit '0' was finished. Besides, the sound for this digit will be played in Arabic and English.

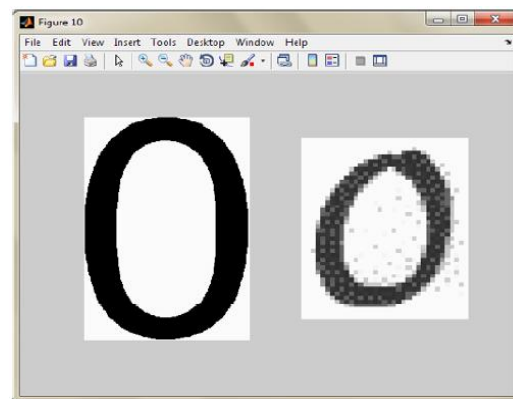


Figure (6): Image detection of the digit '0'.

After the accomplishment of the ANN detection many measurements were obtained. One of these is the regression between the actual output and the target, as presented in figure (7). Another ANN measurement was the mean squared error (MSE) as illustrated in figure (8). In addition, there are other ANN measurements shown in figure (9). This figure shows the Gradient, Validation, and Learning rate against the Epochs.

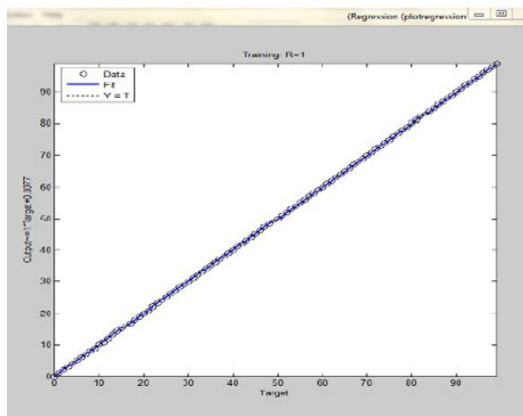


Figure (7): Regression between the actual output and the target.

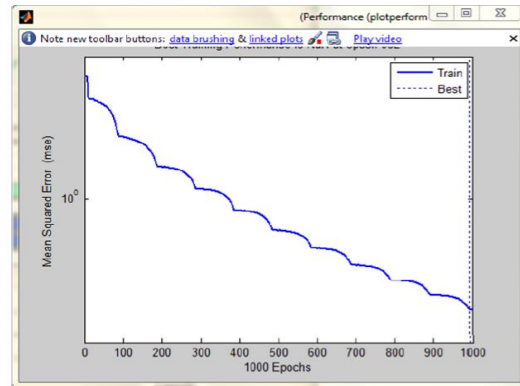


Figure (8): The Mean Squared Error against Epochs.

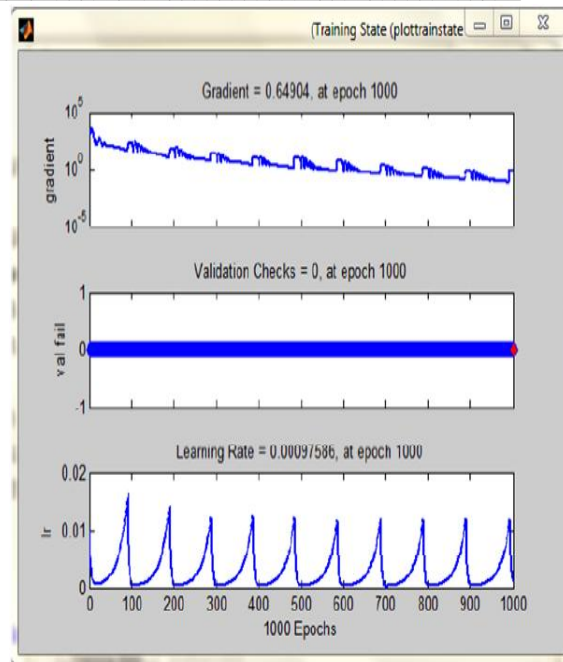


Figure (9): Figure shows Gradient, Validation and Learning Rate against Epochs.

4. CONCLUSION

In the algorithm suggested, for obtaining the detection of the hand writing numbers, this method allowed us a high precision compared with other techniques reported by others, as the turned out shows, this techniques was faster, simpler, efficient, and easy to implement.

The current work can be further investigated to cover many areas:

1: Reading checks in banks; Where a costumer hand-write the numbers when withdraws from his/her account. The scanner or camera will take the image of the check and other programs make split of the whole number into digits, each digits tested by the above suggested work in order to set recognized. This is followed by joining the digits together to produce the whole number. Finally it makes sure that this number is identical

with his/her account before allowing him/her to make withdrawing.

2: This work can also be used in license plate number recognition.

3: This work can be repeated but by using artificial neural networks tuning by a Genetic Algorithm. This network will give us the best trained weights and biases for the networks which will help us to find high precision results in detection the hand writing numbers.

4: Finally, if each hand writing digits has one hundred images of different thickness and shapes instead of ten. Therefore for ten digit numbers from '0' to '9', 1000 cases had resulted to the next stage. This work will be more accuracy and will be reduced the cases of the "Not defined digit".

REFERENCES

- [1] A. Akoum, B. Daya, and P. Chauvet, "Two Neural Networks for License Number Plates Recognition", *Journal of Theoretical and Applied Information Technology*, pp. 25-32, 2009.
- [2] Maarif Haris Al-Qodri and Sar Sardy. "Plate Number Recognition by Using Artificial Neural Network", *proceedings Semiloka Teknologi Simulasi dan Komputasi serta Aplikasi*, pp. 176-182, 2006.
- [3] Hontani H., and Kogth T., "Character extraction method without prior knowledge on size and information", *proceedings of IEEE International Vehicle Electronics Conference (IVECOI)*, pp. 67-72, 2001. [From Iraqi Virtual Science Library, www.ivsl.org].
- [4] Fernando Martin, Maite Garcia, and Jose Luis Alba, "New methods for automatic reading of VLP's (Vehicle License Plates)", 2000.
- [5] Tran Duc Duan, Tran Le Hong Du, Tran VinhPhuoc, and Nguyen Viet Hoang, "Building an automatic vehicle license plate recognition system", *Proceeding International Conference in Computer Science*, pp. 59-63, 2005.
- [6] Yo-Ping Huang, Shi-Yong Lai, and Wei- Po Chuang. "A template based model for license plate ecognition", *proceeding IEEE*, pp. 737-742, 2004.
- [7] Rajeshwar Dass and Seema Asht, "Handwritten Numeral Pattern Recognition Using Neural Network", *International Journal of Computer Science and Telecommunications*, Volume 3, Issue 9, September 2012.
- [8] Firoj Parwej, "English Sentence Recognition using Artificial Neural Network through Mouse-based Gestures", *International Journal of Computer Applications* (0975 – 8887) Volume 61– No.17, pp. 33- 38, January 2013.
- [9] Zurada J.M. *Introduction to artificial neural systems*, West Publishing Company, St. Paul 1992.
- [10] Rafael C. Gonzalez, and Richard E. woods, *Digital Image Processing*. Prentice Hall. New Jersey, 2002.
- [11] Rafael C. Gonzalez, and Richard E. Woods, and Steven L. Eddins, *Digital Image Processing Using Matlab*. Prentice Hall. New Jersey, 2004.