

Offline Signature Verification System Using Artificial Neural Network

Dr. Manoj Kumar Deka

Assistant Professor

Department of Computer Science and Technology,

Bodoland University

Kokrajhar, BTC, Assam-783370

Abstract—Signatures are used every day to authorize the transfer of funds of millions of people. Bank checks, credit cards and legal documents all require signatures. Forgeries in such transactions cost millions of dollars each year. Signature verification is the process carried out to determine whether a given signature is genuine or forged. There are two major methods of signature verification. One is an online method to measure the sequential data such as handwriting and pen pressure with a special device. The other is an offline method that uses an optical scanner to obtain handwriting data written on paper.

Through this paper an attempt has been made to verify offline signature by using the multilayered perceptron neural network technique. The emphasis was on investigating the performance of the neural network when presented with raw signature images using backpropagation algorithm. In this system, the backpropagation neural network algorithm in MATLAB toolbox is used. The performance was evaluated based on a set of 30 signatures. These were split into a training set of 25 samples and a test set 5 signatures. Signatures samples were scanned and processed. Then the neural network is trained to learn and identify whether the signature is genuine or forged.

Keywords-

Backpropagation, Feedforward, Perceptron, Artificial Neural network

I. INTRODUCTION

Biological neural networks are made up of real biological neurons that are connected or functionally related in the peripheral nervous system or the central nervous system. In neuroscience, a biological neural network describes a population of physically interconnected neurons or a group of disparate neurons whose inputs or signalling targets define a recognizable circuit. Communication between neurons often involves an electrochemical process. The interface through which they interact with surrounding neurons usually consists of several dendrites (input connections), which are connected via synapses to other neurons, and one axon (output connection). If the sum of the input signals surpasses a certain threshold, the neuron sends an

action potential (AP) at the axon hillock and transmits this electrical signal along the axon.

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain [1]. It is composed of a large number of highly interconnected processing elements (neurons) working in union 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. Its advantages include [1]:

1. Adaptive learning: An ability to learn how to do tasks based on the data given for training or initial experience.
2. Self-Organization: An ANN can create its own organization or representation of the information it receives during learning time.
3. Real Time Operation: ANN computations may be carried out in parallel, and special hardware devices are being designed and manufactured which take advantage of this capability.
4. Fault Tolerance via Redundant Information Coding: Partial destruction of a network leads to the corresponding degradation of performance. However, some network capabilities may be retained even with major network damage.

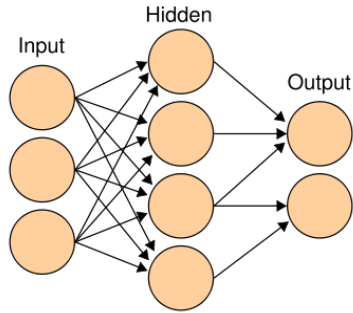


Figure: A multilayer feed forward network

The Digital image processing is the use of computer algorithms to perform image processing on digital images. As a subcategory or field of digital signal processing, digital image processing has many advantages over analog image processing. It allows a much wider range of algorithms to be applied to the input data and can avoid problems such as the build-up of noise and signal distortion during processing. Since images are defined over two dimensions (perhaps more) digital image processing may be modelled in the form of Multidimensional Systems.

Neural backpropagation is the phenomenon in which the action potential of a neuron creates a voltage spike both at the end of the axon (normal propagation) and back through to the dendritic arbor or dendrites, from which much of the original input current originated [2]. It has been shown that this simple process can be used in a manner similar to the backpropagation algorithm used in multilayer perceptrons, a type of artificial neural network.

A multilayer perceptron (MLP) is a feed forward artificial neural network model that maps sets of input data onto a set of appropriate output [1,2]. An MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function. MLP utilizes a supervised learning technique called backpropagation for training the network. MLP is a modification of the standard linear perceptron, which can distinguish data that is not linearly separable.

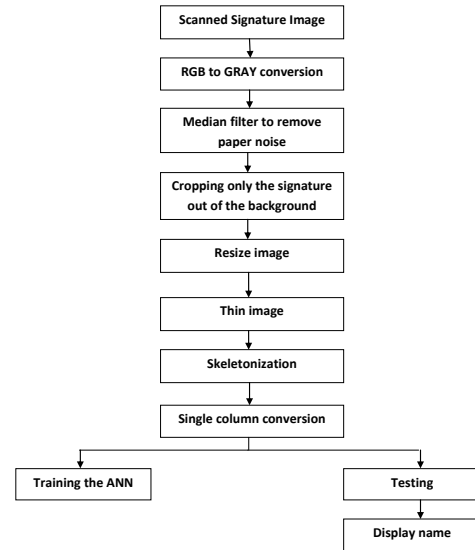


Figure: Block diagram of the system

II. METHODOLOGY AND RESULTS

A. Collection of the Data : Scanned Signature Image:

Create the data base by collecting the signature on white page. For each specimen, six numbers of specific signatures are collected. Then scan them with the help of a digital scanner machine.



Fig:Raw Scanned Signature

B. RGB to GRAY conversion:

Convert RGB image or colormap to gray scale.

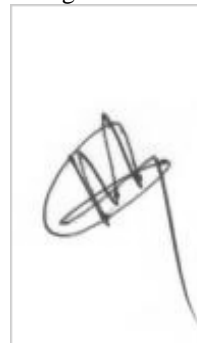


Fig: A typical example of one signature for one specimen when converted from RGB to Gray

C. Binary image conversion:

Binary images contain Boolean pixel values that are either 0 or 1. Pixels with the value 0 are displayed as black; pixels with the value 1 are displayed as white. Intensity images contain pixel values that range between the minimum and maximum values supported by their data type. Intensity images can contain only 0s and 1s, but they are not binary images unless their data type is Boolean. We can use the Relational Operator block to perform a thresholding operation that converts your intensity image to a binary image.

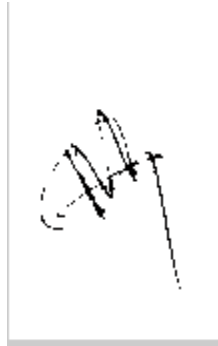


Fig: A typical example of one signature for one specimen when converted from Gray image to binary image

D. Median filter to remove salt and paper noise:

The function `medfilt1` implements one-dimensional median filtering, a nonlinear technique that applies a sliding window to a sequence [3]. The median filter replaces the center value in the window with the median value of all the points within the window [4, 5]. In computing this median, `medfilt1` assumes zeros beyond the input points. When the number of elements n in the window is even, `medfilt1` sorts the numbers, then takes the average of the $(n-1)/2$ and $(n-1)/2 + 1$ elements.

Median filtering is similar to using an averaging filter, in that each output pixel is set to an average of the pixel values in the neighborhood of the corresponding input pixel. However, with median filtering, the value of an output pixel is determined by the median of the neighborhood pixels, rather than the mean. The median is much less sensitive than the mean to extreme values (called outliers). Median filtering is therefore better able to remove these outliers without reducing the sharpness of the image. The `medfilt2` function implements median filtering.

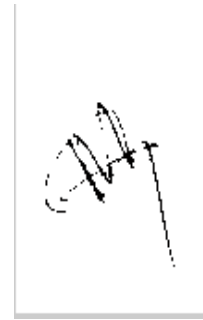


Fig: A typical example of one signature for one specimen when filtered

E. Cropping only the signature out of the background:

Cropping an image means creating a new image from a part of an original image. To crop an image using the Image Tool, use the Crop Image tool.



Fig: A typical example of one signature for one specimen when cropped the background

F. Resize image:

To resize an image, use the `imresize` function. When you resize an image, you specify the image to be resized and the magnification factor. To enlarge an image, specify a magnification factor greater than 1. To reduce an image, specify a magnification factor between 0 and 1.

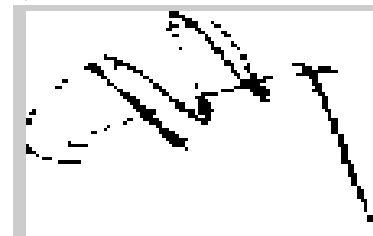


Fig: A typical example of one signature for one specimen when resized

G. Thin image:

$BW2 = \text{bwmorph}(BW, \text{operation})$ [3] applies a specific morphological operation to the binary image BW .



Fig: A typical example of one signature for one specimen when thin image is prepared

H. Skeletonization:

Morphology is a broad set of image processing operations that process images based on shapes. Morphological operations apply a structuring element to an input image, creating an output image of the same size. The most common morphological operation is skeletonization.



Fig: A typical example of one signature for one specimen when skeleton image is prepared

I. Understanding dilation and erosion:

Morphology is a broad set of image processing operations that process images based on shapes. Morphological operations apply a structuring element to an input image, creating an output image of the same size.

J. Single column conversion:

`reshape(x,[m,n])` returns the m-by-n array whose elements are taken by column from x. If x does not have m*n elements, reshape returns an error from the operation [3,4]. Generally, `reshape(x, siz)` returns an n-dimensional array with the same elements as x, but

reshaped to size (siz). Note that `prod(siz)` must be the same as `prod(size(x)).reshape(x,[m,n...])` returns an n-dimensional array with the same number of elements as x, but reshaped to have size m-by-n-by-p-by-.... For the reshape operation to work, `m*n*p*...` must equal `prod(size(x))`.

K. Training the ANN:

Once the network weights and biases are initialized, the network is ready for training. The network can be trained for function approximation (nonlinear regression), pattern association, or pattern classification. The training process requires a set of examples of proper network behavior -- network inputs p and target outputs t. During training the weights and biases of the network are iteratively adjusted to minimize the network performance function `net.performFcn`. The default performance function for feed forward networks is mean square error mse -- the average squared error between the networks outputs a and the target outputs t.

Next you need to train the network to obtain first-layer weights that lead to the correct classification of input vectors.

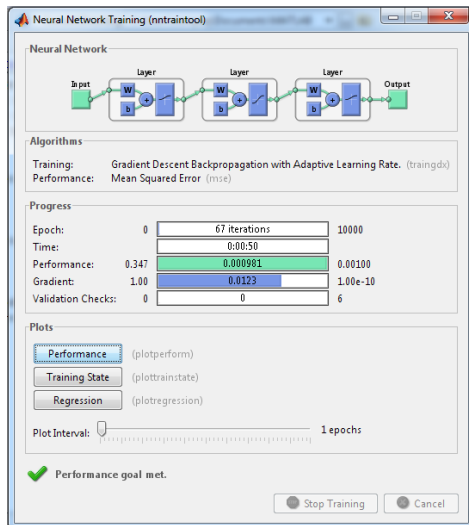


Fig: Training the Artificial Neural network

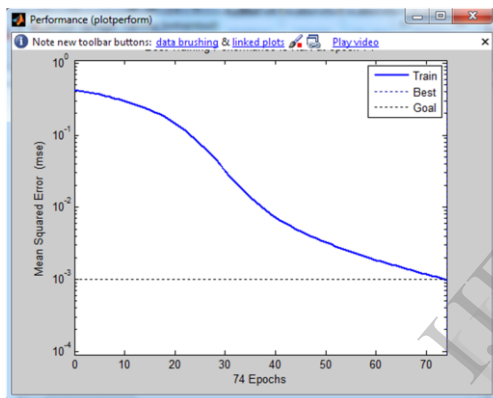


Fig: Performance graph of the Artificial Neural Network

L. Testing:

Open graphical tool for testing MATLAB instrument driver. midtest opens the MATLAB instrument driver testing tool. The MATLAB instrument driver testing tool provides a graphical environment for creating a test to verify the functionality of a MATLAB instrument driver.

The MATLAB Instrument Driver Testing Tool provides a way to [1,3]

- Verify property behaviour
- Verify function behaviour
- Save the test as MATLAB code

- Export the test results to MATLAB workspace, figure window, MAT-file, or the MATLAB Variable Editor
- Save test results as an HTML page

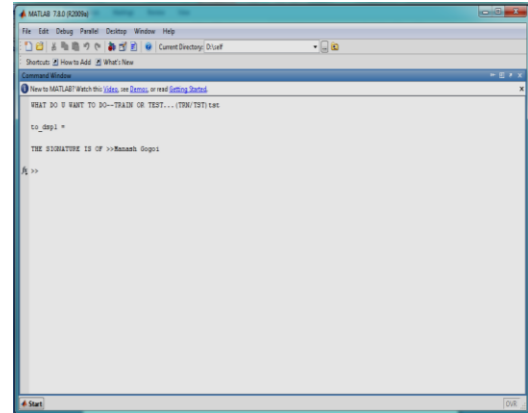


Fig: Verification of signatures

M. Display name:

Display(X) prints the value of a variable or expression, X. MATLAB calls display(X) when it interprets a variable or expression, X, that is not terminated by a semicolon. For example, sin(A) calls display, while sin(A); does not. If X is an instance of a MATLAB class, then MATLAB calls the display method of that class, if such a method exists. If the class has no display method or if X is not an instance of a MATLAB class, then the MATLAB's built-in display function is called.

III. CONCLUSION

The main objective of the system is to verify a signature offline using neural networks. An offline signature verification system using MLP (Multilayer Perceptron) neural network is implemented in this system. The results show that the MLP gives a good accuracy.

IV. REFERENCES

1. N.R. Pal, S.K. Pal, "A Review on Image Segmentation Techniques", "Pattern Recognition"
2. Tong Qu, "Dynamic Signature Verification System Design Using Stroke Based Feature Extraction Algorithm"
3. <http://www.mathworks.com>
4. B. Fang, C.H. Leung, Y.Y. Tang, K.W. Tse, P.C.K. Kwok and Y.K. Wong, "Offline Signature Verification by the Tracking of

- Feature and Stroke Positions", *Pattern Recognition* 36, 2003, pp. 91-101
5. Hertz JA, Palmer RG, Krogh, AS, "Introduction to the Theory of Neural Computation", Addison- Wesley, Redwood City, 1991

IJERT