International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 9, September- 2014

A Study on Signature Verification using Backpropagation Algorithm

Surabhi Jain
BITS – Pilani, Pilani, 333031
Rajasthan

Yashaswy Akella
BITS – Pilani, Hyderabad, 500078
Andhra Pradesh

Sajan Ambadiyil
Center for Development of Imaging Technology
Thiruvananthapuram, 695027
Kerala

V. P. Mahadevan Pillai
Department of Optoelectronics University of Kerala
Thiruvananthapuram 69558
Kerala

***Abstract*—Signature verification is a very challenging area of study. It is not easy to decipher whether a signature is genuine or forged. Various methods have been developed to measure the genuinity of a signature. This paper talks about one such method, the backpropagation algorithm which has the highest accuracy. The sole reason of picking up this method is the ease of use and understanding. Also the ways to implement it in a easier way have been talked about. In the later part, the probable areas of improvement failures have been suggested. These may be improved and the efficiency can be increased in case the study is extended further.**

Keywords : Pattern Recognition, Signature Verification, Artificial Neural Networks, Backpropagation.

I. INTRODUCTION

Signature is socially accepted & extensively used means for authentication in our daily life. It is a very old distinguishing feature for individual's identification. A handwritten signature is the legal mark of the identity of any person, designed on his way of authenticating the documents. Even today, most of the works especially in financial and real estate sectors, offline hand written signatures are preferred. However perfect and difficult the signatures be, they have a threat of getting forged by anyone. Forgery is defined as imitating objects with an intent to deceive. We classify forgery [1] under two categories.

Simple forgery refers to the forger not having access to the original signature and thus forging it in his own style. Skilled forgery is done by artists who has an access to the original signature and is able to produce a deceptive replica of the signature.

Manual signature is very basic method for a person to recognize the signer of the document. It is widely used to recognize a person delivering out daily procedures i.e. bank operations, document analysis, electronic funds transfer, and access control. Biometrics [2] offer automated methods for verification or identification on the principle of measurable physiological or behavioral characteristics such as a signature or a voice sample. These characteristics should not be duplicable, but it is unfortunately often possible to create a copy that is accepted by the biometric system as a true sample. Signature is one of the psychological characteristic of Biometrics. There are two methods [1] to verify the signatures (1) Online (2) Offline. Online method requires special set of devices and instruments to capture the pen movements and pressure over the paper at the same time of the writing. Online data records the motion of the stylus while the signature is produced, and includes the dynamic information such as velocity, acceleration, pen pressures as functions of time. These dynamic characteristics are specific to each individual and sufficiently stable as well as repetitive. Offline, on the other hand, uses an optical scanner in order to obtain a digital representation composed of $M \times N$ pixels. The signature image is considered as a discrete 2D function $f(x, y)$, where x and y denote the spatial coordinates. The value of f in any (x, y) corresponds to the gray level in that point. Processing is done on the scanned images.

The non-repetitive nature of variation of the signatures, because of age, illness, geographic location and perhaps to some extent the emotional state of the person causes large intra-personal variation. Absence of stable dynamic characteristics and uneasiness to segment signature strokes, makes offline signatures difficult to verify. This system has two distinct but strongly related tasks as recognition of the signature & verifying it whether it is genuine or forged. For the verification part, various methods have been studied and the most efficient one has been picked up and is worked upon. This paper analyses the method of signature verification in detail and the probable areas of improvement.

II. GENERALMETHODOLOGY

The following are the steps [3] -

A. Creating the database

Foremost, for the experiment, a Database consisting of all the considered genuine signatures has been created. For the same, the sample space was collected from some chosen individuals whose signatures were to be taken for the experiment and were worked upon. The working involves four different steps – 1.Collection of sample signatures.

2. Preprocessing
3. Feature extraction
4. Storing in the database for reference.

B. Verifying the input signature

To verify, the method used is the Backpropagation algorithm, Artificial Neural Networks. The steps involves –

1.Getting the test signature.

2. Preprocessing
3. Feature Extraction
4. Recognition and verification using backpropagation.
5.

III. STEPSEXPLAINED

A brief explanation of some substeps used for the technique of verification is described below. MATLAB is used to carry out the experiment.

A.Preprocessing

Preprocessing algorithms [4][5] are the data conditioning algorithms for feature extraction process. The purpose is to make signature standard and ready for feature extraction. It improves the quality of the image. It is used to reduce variations and produce a more consistent set of data. Preprocessing should include some noise filtering, smoothing and normalization to correct the image from different errors. Main steps in preprocessing are –

1. Conversion of color image to gray scale – Generally signatures are signed with colored ink and in order to perform the steps in preprocessing, we need to convert the signature image into the gray scale image. Figure below shows the input image and the converted gray scale image.

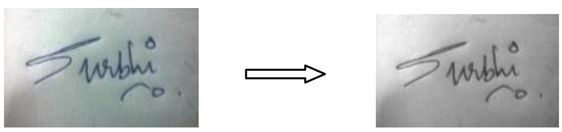
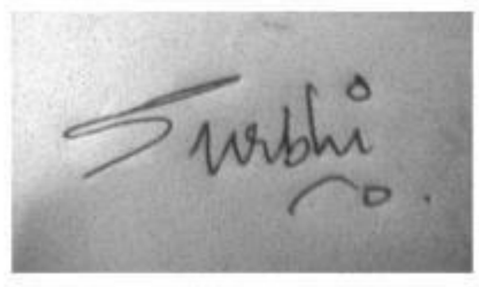
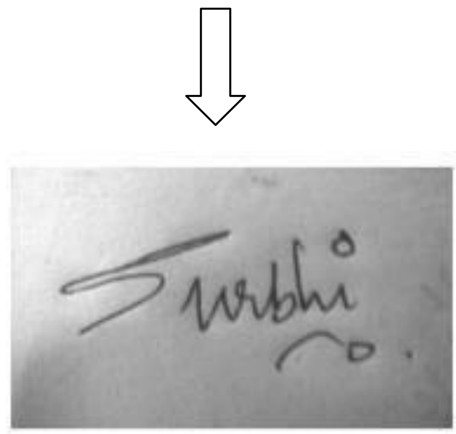


Fig 1 (a) input image (b) gray scale image.

2. Noise Reduction [5] – Median filter is widely used for smoothing and restoring images corrupted by noise because of the fact that it reduces impulsive or salt-and-pepper type noise and has attractive properties for suppressing impulsive noise while preserving edges. Salt and pepper is added because there may be small disturbances which may not be detected while removing noise. To avoid this and remove noise from the image salt-and –pepper is added and then the small disturbances are removed along with the added noise.



(a) Salt and Pepper is added



(b) Final denoised image.

Fig 2 (a) Nosed image with added salt and pepper
(b) Denoised Image

3. Background elimination and border clearing – Thresholding [6][5] is the easiest and sophistically applicable method for background elimination and border clearing. A threshold value is chosen and all the pixels having values less than the threshold value is assigned zero and greater than as one. The high pixel intensity value is the background and hence the signature is separated from the background. The obtained signature image is then converted into binary image and vertical and horizontal projections are used for border clearing.
Figure 3 shows the result obtained for background elimination. It has three axes – x, y, z along with the separated signature.

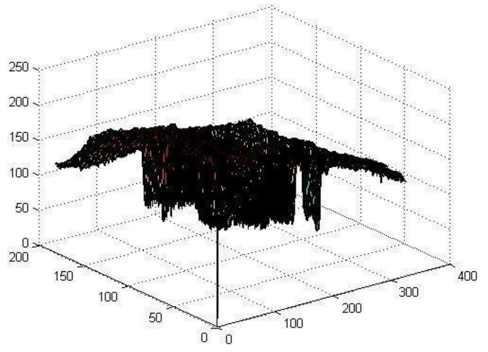


Fig 3 Background Elimination

4. Normalization - Signature dimensions may vary due to irregularities in image scanning and capturing process. Furthermore height and width, size of the signature, may vary from signature to signature. It is thus normalized. During normalization, the ratio of width to height is kept intact. Figure 4 shows the final image obtained after normalization.

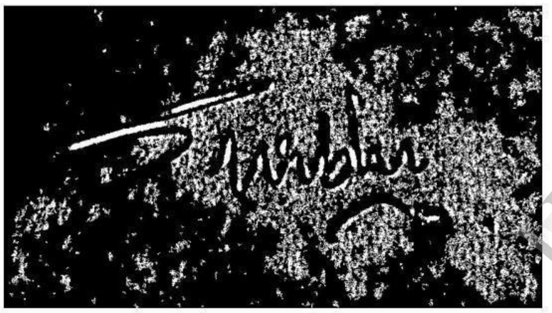


Fig 4 Final Image obtained after Normalization

B.Feature Extraction [6]

Extracting the information from the raw data which is most relevant for classification stage is feature extraction. This data can be minimized within-class pattern variation and increases inter-class variations. Feature extraction is a special form of dimensionality reduction. More than 40 different feature types have been used for signature verification. Features extracted should be easily computed, robust, rotationally invariant, and insensitive to various distortions and variations in the images. The features that are extracted in this phase are used to create a feature vector.

Figure 5 shows various MSER features extracted using MATLAB.

The green color shows the Features extracted in the signature.

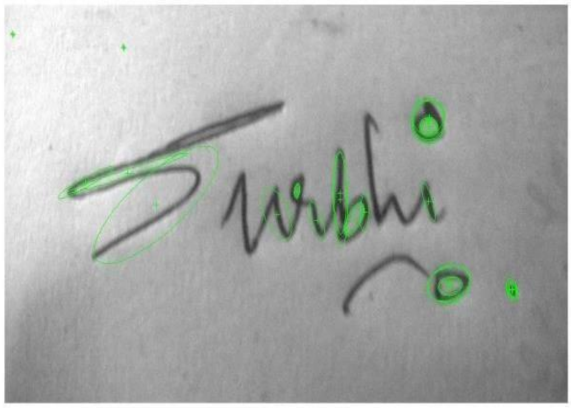


Figure 5

C.Verification

A feature vector including all the feature points is created. Extracted feature points from all the signatures are normalized to the range of 0 to 1. These normalized features are applied as an input to the neural network and the network is trained. Taking the test signature and making it go through exactly the same steps as preprocessing, feature extraction etc, it is put into the trained neural network and the result is calculated using backpropagation algorithm.

IV. ARTIFICIAL NEURAL NETWORKS[7][8]

(A) *Networks*

An efficient way to solve a complex problem is to decompose it into various simple problems and find out solutions for them individually. Networks work exactly the same way. There are a large number of different types of networks but all are characterized by – a set of nodes and the connection between nodes. Nodes are computational units. They receive inputs, and process them to obtain an output. The processing may be as simple as doing arithmetic or as complex as making an altogether different network out of the existing one.

One type of network sees the nodes as ‘artificial neurons’. They are called Artificial Neural Network (ANN). These are computational models inspired by an animal's central nervous systems (in particular the brain) which is capable of machine learning as well as pattern recognition. Artificial neural networks are generally presented as systems of interconnected "neurons" which can compute values from inputs. Artificial neural network types vary from those with only one or two layers of single direction logic, to complicated multi–input many directional feedback loops and layers. On the whole, these systems use algorithms in their programming to determine control and organization of their functions. An ANN is typically defined by three types of parameters:*

- The interconnection pattern between the different layers of neurons.
- The learning process for updating the weights of the interconnections

- The activation function that converts a neuron's weighted input to its output activation.

The complexity of real neurons is highly abstracted when modelling artificial neurons. They consist of inputs like synapses and weights which are the strength of the respective signals and then are computed by a mathematical function which determines the activation of the neuron. Another function computes the output of the artificial neuron, sometimes in dependence of a certain threshold. ANNs combine artificial neurons in order to process information.

(B) *BACKPROPAGATION ALGORITHM*

Training a neural network model essentially means selecting one model from the set of allowed models that minimizes the error. There are numerous algorithms available for training neural network models. Most of the algorithms used in training artificial neural networks employ some form of gradient descent. This is done by simply taking the derivative of the error function with respect to the network parameters and then changing those parameters in a gradient-related direction.

The back propagation algorithm has its artificial neurons organized in layers, and send their signals "forward", and then the errors are propagated backwards. The network receives inputs by neurons in the input layer, and the output of the network is given by the neurons on an output layer. There may be one or more intermediate hidden layers. The back propagation algorithm uses supervised learning which means that we provide the algorithm with examples of the inputs and outputs we want the network to compute, and then the error (difference between actual and expected results) is calculated. In other words, we wish to infer the mapping implied by the data.

OVERVIEW OF THE ALGORITHM

1. Choose random weights for the network
2. Feed in an example and obtain a result.
   The activation function of the artificial neurons in ANNs implementing the back propagation algorithm is –
   $$A_j(\bar{x},\bar{w}) = \sum_{i=0}^{n} x_i w_{ji} \quad \ldots\ldots\ldots (1)$$
   Where 'x' is the input and 'w' is the weight of the node. We can see that the activation depends only on the inputs and the weights.
   The output function is the sigmoidal function.
   $$O_j(\bar{x},\bar{w}) = \frac{1}{1+e^{A_i(\bar{x},\bar{w})}} \quad \ldots\ldots\ldots (2)$$
   We can see that the output depends only on the activation, which in turn depends on the values of the inputs and their respective weights.
3. Calculate the error for each node, starting from the last stage and propagating the error backwards. The goal of the training process is to obtain a desired output when certain inputs are given. Since the error is the difference between the actual and the desired output, the error depends on the weights, and we need to adjust the weights in order to minimize the error. A commonly used method is the mean-squared error, which tries to minimize the average squared error between the network's output, and the target value over all the example pairs. We can define the error function for the output as
   $$E_j(\bar{x},\bar{w},d) = (O_j(\bar{x},\bar{w}) - d_j)^2 \quad \ldots\ldots (3)$$

Minimizing this error using gradient descent for the class of neural networks is called well known back propagation algorithm for training neural networks.

4. Update the weights as –
   $$\Delta w_{ji} = -\eta \frac{\partial E}{\partial w_{ji}} \quad \ldots\ldots (4)$$
5. Repeat with other examples until the network converges on the target output.

V. PROBABLE IMPROVEMENTS

There is no method that is more proper and accurate than the neural networks back propagation as it achieves the maximum accuracy yet it has some demerits which we couldn't ignore.

1. When using the back-propagation to train a standard multi-layer neural network, the designer is required to arbitrarily select parameters such as the network topology, initial weights and biases, a learning rate value, the activation function, and a value for the gain in the activation function. Improper selection of any of these parameters can result in slow convergence or even network paralysis where the training process comes to a virtual standstill.
2. Single evaluation of error function for a single input requires O(W) operations for large W. Therefore, for practical reasons, ANNs implementing the back propagation algorithm do not have too many layers, since the time for training the networks grows exponentially as the output function is a sigmoid function. Refer to Equation 2.
3. Also, the error function, as in Eqution 3 is calculated using the mean squared method to achieve more accurate results and Equation 4 is a gradient descent method so with the tendency of error function to have the steepest descent technique, which is used in the training process, one can easily get stuck at local minima.
   Gradient descent is based on the observation that if the multivariable function F(x) is defined and differentiable in a neighborhood of a point 'a' then F(x) decreases fastest if one goes from 'a' in the direction of negative gradient of F at 'a'. So, the convergence to the local minimum is guaranteed in this case though convergence to global minimum depends on the function being convex or concave.

VI. CONCLUSION

Since there is no method which has 100% accuracy, the research is still going on in this area to achieve maximum accuracy and to improve the back propagation algorithm. Some of the methods may include the refinement of the gradient descent method. The second improvement may direct towards the randomness in choosing the parameters such as network topology, initial weights and biases, a learning rate value, the activation function, and a value for the gain in the activation function. It has been found out that changing the weight value adaptively for each node can significantly reduce the training time.

VII. REFERENCES

1. Approaches and Issues in Offline Signature Verification system, Hemanta Saikia and Kanak Chandra Sarma International Journal of Computer Applications (0975 – 8887) Volume 42– No.16, March 2012
2. S. Rajasekaran and G.A. Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic, and Genetic Algorithms", Prentice Hall India Pvt Signature Recognition using Artificial Neural Network – Debanath Bhattacharya and Tai-hoon kim Science and Engineering Research Support Center, Republic of Korea.
Department of Multimedia Engineering, Hannam University.
3. In Jain, AK; Flynn; Ross, A. Handbook of Biometrics
4. Digital Image Processing, Rafeal C Gonzalez, Richard E Woods. Addison-Wesley Publishing Company, printed in September 1993
5.http://www.eng.iastate.edu/ee528/sonkamaterial/chapter_4_3.htm
6. http://www.indianjournals.com/ijor.aspx?target=ijor:wea&volume=18&issue=4&article=abs051
7. A. Pacut, A Czajka, "Recognition of Human Signatures"
8. W. Nelson, W.Turin, T.Hastie, "statistical methods for online signature verification. International journal of Pattern Recognition and Artificial Intelligence, 8, 1994.
9. An Introduction to Artificial Neural Networks – Jaico Publishing House
10. Gonzalez, Rafael C. & Woods, Richard E. (2002). Thresholding. In Digital Image Processing, pp. 595–611. Pearson Education
11. Artificial Neural Networks – Theory and Applications. Dan W Patterson 1996 by prentice hall.