Summer Industrial Training Project Report
June–July 2019

Static Signature Recognition using Neural Network

S Rohit
Heritage Institute of Technology

Abstract

Of all biometric techniques, handwritten signature remains the mostly widely used method in legal, finance and social settings for authenticating a document or a person's identity. Therefore, the need for an automatic system for signature recognition and verification is critical.

Approach: After pre-processing the signature into a binary image of preset dimensions with proper alignment, its topological and texture patterns are extracted and fed into an Artificial Neural Network.

1. **Signature Identification** *("Who could the signature belong to?")*: With proper training, the neural network was able to identify users with 98% accuracy.
2. **Signature Verification** *("Is this signature genuine?")*: With the right acceptance threshold, the neural network was able to correctly detect forgeries with 70% accuracy.

Keywords: Image Processing, OpenCV; Biometrics: Signature recognition, identification, authentication, verification; Artificial Neural Network; Backpropagation with momentum

1 INTRODUCTION

Biometrics is a measurable characteristic of an individual which can be used to uniquely identify him/her. Unlike traditional techniques (passwords, PIN, smartcards, etc), biometrics cannot be stolen, lost, forgotten, or shared, and so, have a growing role in the Internet and society.

Signature recognition is a behavioural biometric which can be performed in two ways:

1. *Static ("off-line")*: users write their signature on paper, digitize it with a camera, and the system recognizes it by analyzing its shape.
2. *Dynamic ("on-line")*: users write their signature on a phone, tablet, PDA, etc - which acquires the signature in real time while keeping track of dynamic information like velocity, pen pressure, azimuth, inclination, etc.

Static method is more common and widely used in legal, finance and social situations because of its simplicity, low cost and the fact that the individual need not be present at the time of verification. On the other hand, dynamic method is more accurate because it takes more factors into consideration.

Method The various algorithms differ mainly in their analysis phase; but they all involve:

- *Data Acquisition*: the signature must be in the required digital format.
- *Pre-processing*: the image must be normalized, resized to proper dimensions, rotated and thinned, background noise removed, etc.
- *Feature Extraction*: a set of various measurable discrete attributes of the signature is extracted.
- *Enrollment & Training*: the extracted features along with information about the user is saved to a database and analysis begins.
- *Evaluation*: test data is pre-processed, features extracted, and analyzed similarly to find likeness to any signature in the database.

About the project The project aims to perform static signature recognition with Neural Network, trained using BackPropagation with momentum.

Github: `srohit-HITK/SignatureRecNN.git`

2 SYSTEM DESIGN

The system takes a (digital) signature as input and performs the following three main steps.

2.1 Pre-processing

- **Black-n-White:** to remove irrelevant information like color, brightness, noise, etc.

  > Read the signature as a grayscale (single channel) image and convert it to black-and-white binary with a threshold of 200

- **Rotation angle:** to make the axes depend only on inherent attributes of the signature (viz center of gravity) and not on any external factor like the angle at which the image was taken.

  > Let S_R be the median points of pixels per column and L_R be the regression line over the set S_R. Rotate the image to make L_R the new X-axis

- **Crop & Scale:** to limit our analysis to the required region (bounding box) and normalize the dimensions for feature extraction.

2.2 Feature Extraction

F1. Height-width ratio of the cropped un-scaled image; usually constant for a given individual.

F2. Ink ratio: percentage of black-to-white pixels in the cropped, scaled image.

F3. Pixel count: the number of pixels in (a) every column and (b) every row of the image.

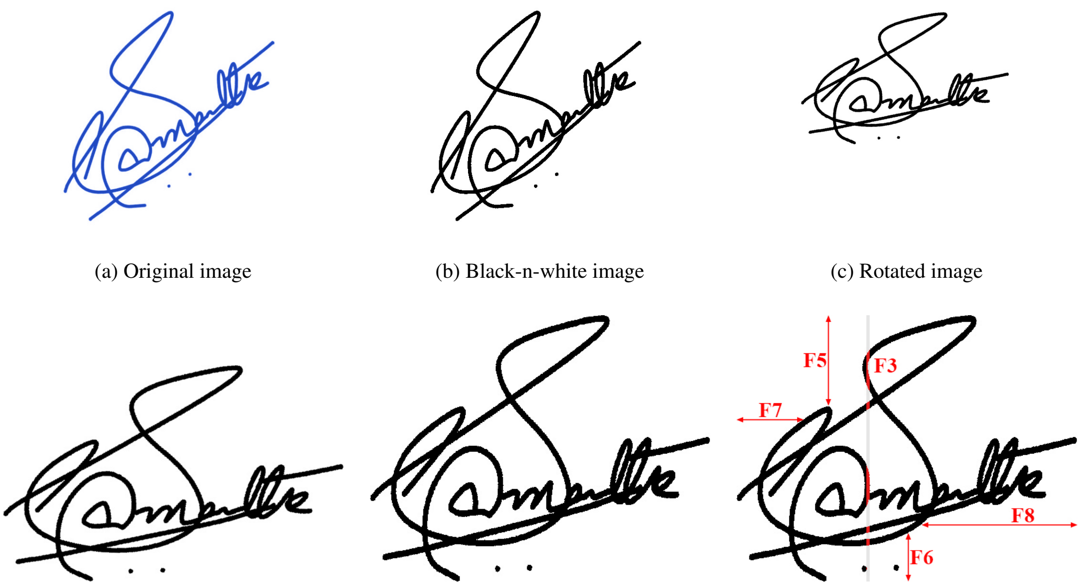
F4. Mean points: the mean of the pixel positions of (a) every column and (b) every row.

F5. Upper envelope: the position of first pixel in every column of the image.

F6. Lower envelope: the position of the last pixel in every column of the image.

F7. Left envelope: the position of the first pixel in every row of the image.

F8. Right envelope: the position of the last pixel in every row of the image.



(a) Original image (b) Black-n-white image (c) Rotated image

(d) Cropped image (e) Scaled image (f) Image features

Figure 1: Pre-processing and feature extraction

2.3 Artificial Neural Network (ANN)

- Input layer has approx 100 neurons. The feature vector was 'normalized' to make it independent of image dimensions and then it was reduced to 100 by taking medians of sets of 8.
- Output layer has as many neurons as the number of registered users in the signature database.
- Only 1 hidden layer of 30 nodes. A simple setup was preferred for this demo.
- Learning rate, $\eta = 0.3$ and momentum, $\alpha = 0.1$
- $30k$ iterations per round unless MSE $\leq$ 1E-6

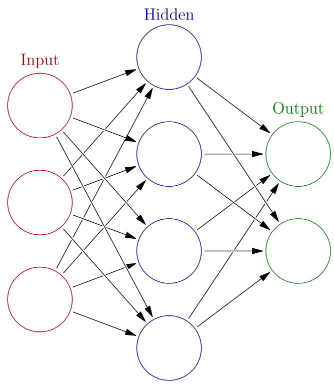


Figure 2: Artificial Neural Network

Feed Forward Activation value $o_i^{(\ell)} = \sigma(\mathrm{net}_i)$ for all neurons, i in all layers ℓ except input-layer where $\mathrm{net}_i = \sum_j w_{ij}^{(\ell)}\, o_j^{(\ell-1)}$ and activation fn, $\sigma(x)$ taken as sigmoid curve $[1+\exp(-x)]^{-1}$

Backpropagation with momentum Weight adjustment, $\Delta w_{ij}^{(\ell)} = -\eta \dfrac{\partial E}{\partial w_{ij}^{(\ell)}} + \alpha \Delta w_{ij}^{(\ell-1)}$

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial o_j}\frac{\partial o_j}{\partial \mathrm{net}_j}\frac{\partial \mathrm{net}_j}{\partial w_{ij}} = o_i \delta_j$$

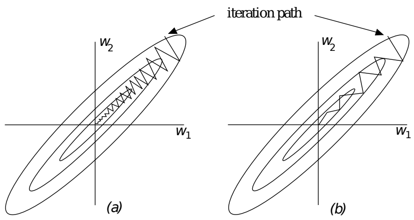


Figure 3: Momentum accelerates the learning process

Brief Derivation:

$$\frac{\partial \mathrm{net}_j}{\partial w_{ij}} = \frac{\partial}{\partial w_{ij}} \sum_k w_{kj} o_k = (0 + \cdots + o_i + \cdots + 0) = o_i$$

$$\frac{\partial o_j}{\partial \mathrm{net}_j} = \frac{\partial}{\partial \mathrm{net}_j}\sigma(\mathrm{net}_j) = \sigma(\mathrm{net}_j)(1-\sigma(\mathrm{net}_j)) = o_j(1-o_j)$$

$$\frac{\partial E}{\partial o_j} = \frac{\partial}{\partial o_j}\frac{1}{2}(t-y)^2 = y - t, \quad \text{if } j \text{ is an output neuron, ie. } o_j = y$$

$$\begin{aligned}\frac{\partial E}{\partial o_j} &= \sum_{\ell \in L} \frac{\partial E}{\partial \mathrm{net}_\ell}\frac{\partial \mathrm{net}_\ell}{\partial o_j}, \quad \text{if } j \text{ is an inner neuron.} \\ &= \sum_{\ell \in L} \frac{\partial E}{\partial o_\ell}\frac{\partial o_\ell}{\partial \mathrm{net}_\ell}\frac{\partial \mathrm{net}_\ell}{\partial o_j} \\ &= \sum_{\ell \in L} \frac{\partial E}{\partial o_\ell}\frac{\partial o_\ell}{\partial \mathrm{net}_\ell} w_{j\ell}\end{aligned}$$

$$\text{Let } \delta_j = \frac{\partial E}{\partial o_j}\frac{\partial o_j}{\partial \mathrm{net}_j} = \begin{cases} (o_j - t_j)\ o_j(1-o_j) & \text{if } j \text{ is an output neuron,} \\ \sum_{\ell \in L} w_{j\ell}\delta_\ell\ o_j(1-o_j) & \text{if } j \text{ is an inner neuron.}\end{cases}$$

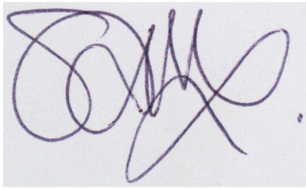
3 EXPERIMENTAL RESULTS

Data Set The ICFHR 2012 Signature Verification Competition (`4NSigComp2012`) database provided 15 genuine and 449 test signatures for 3 individuals. The test signatures were a mix of genuine signatures as well as several skilled and amateur forgeries.

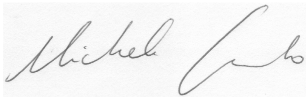
Result The system correctly recognized all 440 signatures with an average 'confidence' of 95%

Ind	*Theshold*	*FRR*	*FAR*	*Accuracy*
A1	99.20%	21.22%	6.94%	71.84%
A2	97.07%	13.54%	20.83%	65.63%
A3	99.60%	25.25%	0%	74.75%

1. Avg False Rejection Rate (FRR): 20%
2. Avg False Acceptance Rate (FAR): 9%
3. Avg Accuracy (Verification): 70%
4. Accuracy (Identification): 98%



Specimen A1



Specimen A2

Specimen A3

Figure 4: Samples from the data set

4 ACKNOWLEDGMENT

I would like to thank *Mr. Upendra Roy and Sha Infotech* for the invaluable guidance and unwavering support, without which I would not have been able to complete this project.

REFERENCES

[1] Rituparna Datta Debaleena Jana, Ranjan Saha. Offline signature verification using euclidian distance. *International Journal of Computer Science and Information Technologies*, 5(1):707–710, 2014.

[2] Sargur Xu Aihua Kalera, Meenakshi K Srihari. Offline signature verification and identification using distance statistics. *International Journal of Pattern Recognition and Artificial Intelligence*, 18(07):1339–1360, 2004.

[3] Sanjivani Pandey, Vibha Shantaiya. Signature verification using morphological features based on artificial neural network. *International Journal of Advanced Research in Computer Science and Software Engineering*, 2(7), 2012.

[4] Guy Plamondon, Rejean Lorette. Automatic signature verification and writer identification—the state of the art. *Pattern recognition*, 22(2):107–131, 1989.

[5] Ashwini Pansare Shalini Bhatia. Handwritten signature verification using neural networks. *International Journal of Computer Applications*, Volume 1(2), 2012.

[6] Minal Kashid Chetana Sthapak, Shiwani Khopade. Artificial neural network based signature recognition & verification. *International Journal of Emerging Technology and Advanced Engineering (IJETAE)*, 2(8):191–197, 2013.

[7] H B Kekre V A Bharadi. Off-line signature recognition systems. *International Journal of Computer Applications (0975 - 8887)*, Volume 1(27), 2010.