Signature Verification using a "Siamese" Time Delay Neural Network

Jane Bromley, Isabelle Guyon, Yann LeCun,
Eduard Săckinger and Roopak Shah
AT&T Bell Laboratories
Holmdel, NJ 07733
jbromley@big.att.com

Abstract

This paper describes an algorithm for verification of signatures written on a pen-input tablet. The algorithm is based on a novel, artificial neural network, called a "Siamese" neural network. This network consists of two identical sub-networks joined at their outputs. During training the two sub-networks extract features from two signatures, while the joining neuron measures the distance between the two feature vectors. Verification consists of comparing an extracted feature vector with a stored feature vector for the signer. Signatures closer to this stored representation than a chosen threshold are accepted, all other signatures are rejected as forgeries.

1 INTRODUCTION

The aim of the project was to make a signature verification system based on the NCR 5990 Signature Capture Device (a pen-input tablet) and to use 80 bytes or less for signature feature storage in order that the features can be stored on the magnetic strip of a credit-card.

Verification using a digitizer such as the 5990, which generates spatial coordinates as a function of time, is known as dynamic verification. Much research has been carried out on signature verification. Function-based methods, which fit a function to the pen trajectory, have been found to lead to higher performance while parameter-based methods, which extract some number of parameters from a signature, make a lower requirement on memory space for signature storage (see Lorette and Plamondon (1990) for comments). We chose to use the complete time extent of the signature, with the preprocessing described below, as input to a neural network, and to allow the network to compress the information. We believe that it is more robust to provide the network with low level features and to allow it to learn higher order features during the training process, rather than making heuristic decisions e.g. such as segmentation into balistic strokes. We have had success with this method previously (Guyon *et al.*, 1990) as have other authors (Yoshimura and Yoshimura, 1992).

2 DATA COLLECTION

All signature data was collected using 5990 Signature Capture Devices. They consist of an LCD overlayed with a transparent digitizer. As a guide for signing, a 1 inch by 3 inches box was displayed on the LCD. However all data captured both inside and outside this box, from first pen down to last pen up, was returned by the device. The 5990 provides the trajectory of the signature in Cartesian coordinates as a function of time. Both the trajectory of the pen on the pad and of the pen above the pad (within a certain proximity of the pad) are recorded. It also uses a pen pressure measurement to report whether the pen is touching the writing screen or is in the air. Forgers usually copy the shape of a signature. Using such a tablet for signature entry means that a forger must copy both dynamic information and the trajectory of the pen in the air. Neither of these are easily available to a forger and it is hoped that capturing such information from signatures will make the task of a forger much harder. Strangio (1976), Herbst and Liu (1977b) have reported that pen up trajectory is hard to imitate, but also less repeatable for the signer. The spatial resolution of signatures from the 5990 is about 300 dots per inch, the time resolution 200 samples per second and the pad's surface is 5.5 inches by 3.5 inches. Performance was also measured using the same data treated to have a lower resolution of 100 dots per inch. This had essentially no effect on the results.

Data was collected in a university and at Bell Laboratories and NCR cafeterias. Signature donors were asked to sign their signature as consistently as possible or to make forgeries. When producing forgeries, the signer was shown an example of the genuine signature on a computer screen. The amount of effort made in producing forgeries varied. Some people practiced or signed the signature of people they knew, others made little effort. Hence, forgeries varied from undetectable to obviously different. Skilled forgeries are the most difficult to detect, but in real life a range of forgeries occur from skilled ones to the signatures of the forger themselves.

Except at Bell Labs., the data collection was not closely monitored so it was no surprise when the data was found to be quite noisy. It was cleaned up according to the following rules:

- Genuine signatures must have between 80% and 120% of the strokes of the first signature signed and, if readable, be of the same name as that typed into the data collection system. (The majority of the signatures were donated by residents of North America, and, typical for such signatures, were readable.) The aim of this was to remove signatures for which only some part of the signature was present or where people had signed another name e.g. Mickey Mouse.

- Forgeries must be an attempt to copy the genuine signature. The aim of this was to remove examples where people had signed completely different names. They must also have 80% to 120% of the strokes of the signature.
- A person must have signed at least 6 genuine signatures or forgeries.

In total, 219 people signed between 10 and 20 signatures each, 145 signed genuines, 74 signed forgeries.

3 PREPROCESSING

A signature from the 5990 is typically 800 sets of x, y and pen up-down points. $x(t)$ and $y(t)$ were originally in absolute position coordinates. By calculating the linear estimates for the x and y trajectories as a function of time and subtracting this from the original x and y values, they were converted to a form which is invariant to the position and slope of the signature. Then, dividing by the y standard deviation provided some size normalization (a person may sign their signature in a variety of sizes, this method would normalize them). The next preprocessing step was to resample, using linear interpolation, all signatures to be the same length of 200 points as the neural network requires a fixed input size. Next, further features were computed for input to the network and all input values were scaled so that the majority fell between $+1$ and -1. Ten different features could be calculated, but a subset of eight were used in different experiments:

feature 1 pen up = -1 ; pen down = $+1$, (pud)

feature 2 x position, as a difference from the linear estimate for $x(t)$, normalized using the standard deviation of y, (x)

feature 3 y position, as a difference from the linear estimate for $y(t)$, normalized using the standard deviation of y, (y)

feature 4 speed at each point, (spd)

feature 5 centripetal acceleration, (acc-c)

feature 6 tangential acceleration, (acc-t)

feature 7 the direction cosine of the tangent to the trajectory at each point, ($\cos\theta$)

feature 8 the direction sine of the tangent to the trajectory at each point, ($\sin\theta$)

feature 9 cosine of the local curvature of the trajectory at each point, ($\cos\phi$)

feature 10 sine of the local curvature of the trajectory at each point, ($\sin\phi$)

In contrast to the features chosen for character recognition with a neural network (Guyon *et al.*, 1990), where we wanted to eliminate writer specific information, the features such as speed and acceleration were chosen to carry information that aids the discrimination between genuine signatures and forgeries. At the same time we still needed to have some information about shape to prevent a forger from breaking the system by just imitating the rhythm of a signature, so positional, directional amd curvature features were also used. The resampling of the signatures was such as to preserve the regular spacing in time between points. This method penalizes forgers who do not write at the correct speed.