A NEW APPROACH TO SIGNATURE VERIFICATION: DIGITAL DATA ACQUISITION PEN

Ondřej Rohlík, Pavel Mautner, Václav Matoušek, Jürgen Kempf
University of West Bohemia, Faculty of Applied Sciences
Univerzitní 22, CZ – 306 14 Plzeň (Pilsen), Czech Republic
Tel: +420 19 7491 184, Fax: +420 19 7491 213, E-mail: {rohlik mautner matousek}@kiv.zcu.cz
University of Technology (Fachhochschule) Regensburg,
Prüfeninger Straße 58, D – 93049 Regensburg, Germany
Tel: +49 941 943 1270, Fax: +49 941 943 1426, E-mail: juergen.kempf@mikro.fh-regensburg.de

Abstract: *This paper presents our experience with a completely new approach to handwritten text recognition. A brief description of a new type of input device is followed by a more detailed explanation of recognition methods used. Results achieved are discussed and ideas for further research are suggested.*

KEYWORDS: signature verification, person identification, character recognition, OCR, HCI

1 Introduction

There are many commercial systems designed for person identification worldwide. Among the most popular rank those based on fingerprints, ID cards and signature recognition using optical character recognition (OCR). The current systems are based on input devices that consist of at least two parts (pen and tablet, pen with infrared transmitter and one or two receivers, etc.). The obvious problem of such an approach is the limited mobility of a system composed of several parts. Moreover, the cost of such a system is higher than the cost of a system built on the all-in-one principle.

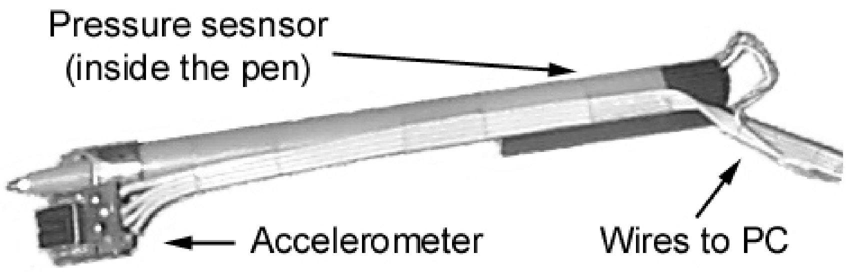

Fig. 1 – Acceleration sensor pen

2 Background

2.1 Data acquisition device

The first experimental pen was built at University of Technology Regensburg[1] during the spring of 2000 (Fig. 1). It consists of two sensors integrated in a pen producing a total of three signals (Fig. 2). The acceleration sensor – accelerometer placed near the nib produces two signals

[1] The development and related research is funded by the bilateral project No. TSR-007-99/ME363

corresponding to the horizontal and vertical movements of the pen. A pressure sensor – based on the piezoelectric effect – is built into the pen (see the two thin wires on the right-hand side in Fig. 1).

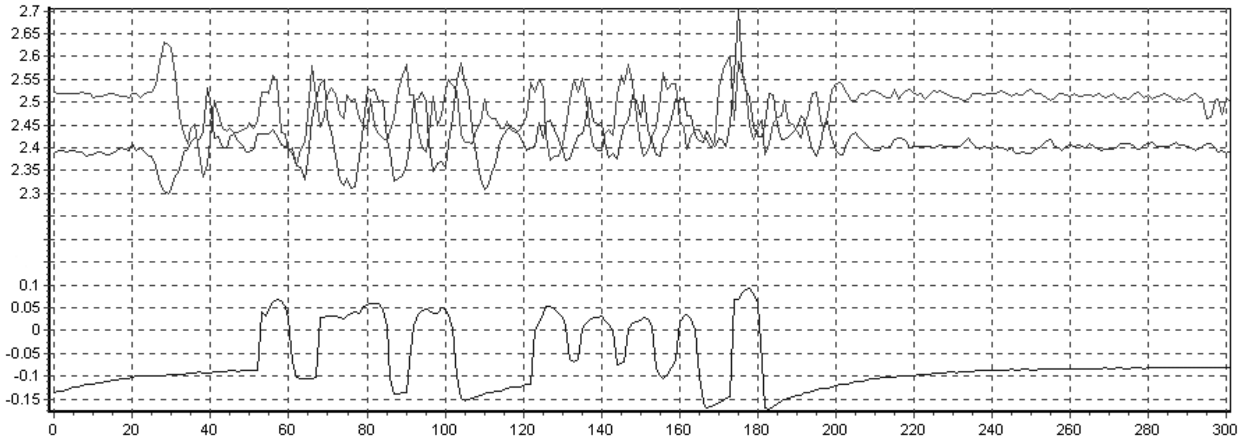

Fig. 2 – Signals of acceleration and pressure

The accelerometer used is an ordinary commercial integrated circuit made by Analog Devices. ADXL202 is 2-axis acceleration sensor that uses a mass that moves between the two pairs of capacitors. The sensor output signal is derived from the change in capacity. The two pairs of capacitors are orthogonal another to one, so the obtained signals correspond to the acceleration x and y axes (two signals with high values in Fig. 2). In fact, this is true if the pen is held in the proper position (Fig. 3). If the pen is slightly rotated, then the x axis does not correspond exactly to the direction of writing (direction of lines); the y axis does not correspond to the direction orthogonal to the line. This problem is solved by working with polar coordinates (amplitude and angle) computed from the x and y signals.
The accelerometer is able to measure both the dynamic acceleration (movement of pen, vibration) and static acceleration (gravity). As the measurement of static acceleration cannot be avoided we have to eliminate the influence of the gravity on the data.

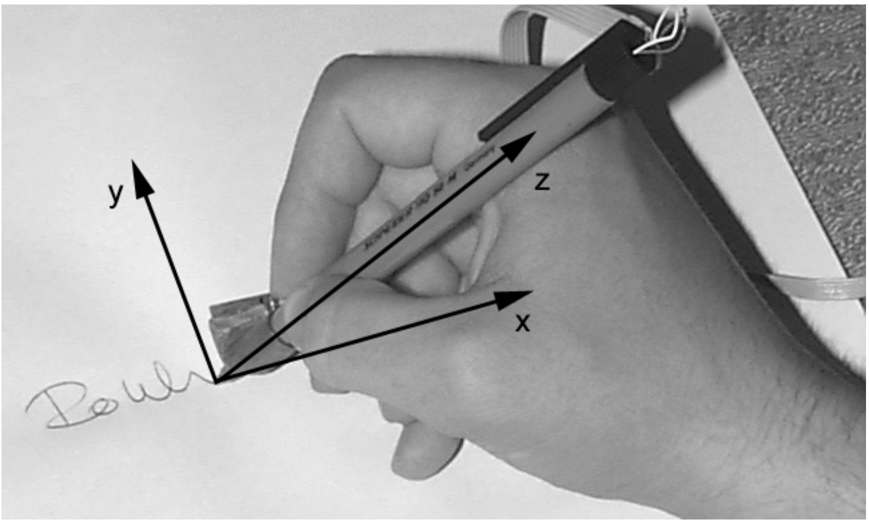

Fig. 3 – Writing with pen – ideal holding

The accelerometer is not designed to be used in handwritten text applications and therefore the signals produced are not suitable for character recognition. This problem is caused by nonlinear conversion of the mass movement to voltage.

We carried out a number of experiments trying to reconstruct the trajectory of the pen by computing the velocity and position but the results were poor. This works only if the signatures (or characters) are very large (as large as a sheet of paper) and even if they are so large, results are not very good.
The pressure sensor used – the PSt150/2x3/7 by Piezomechanik GmbH produces much better signals. We can easily identify when the pen was in contact with the paper and when not. For example the signal with the lowest value in Fig. 2 corresponds to the signature with five separate parts, where the fourth part is the longest and consists of four characters. After the end of each of the five parts there is a slight decrease in voltage, caused by the capacitor added to the pressure signal in order to stabilize it.
Diacritical marks as well as parts of the signature where the trajectory changes radically can be recognized. The pressure signal is extremely valuable when structural recognition methods are used because primitives can be easily identified. A recognizer using these primitives is now under development and the first results are expected at the end of August.

2.2 Application areas

There are three application areas for our pen. The first one – signature verification will be described in detail in this paper. The main task is to find out whether the signature is authentic or fake. Another possible application is person identification. The task is to recognize which one of several people have written the test word assuming that the recognizer has a set of training words from each author. The last possible application of the pen developed is character/text recognition. If the problem of text recognition is solved, keyboards could by replaced by the pen. Advantages of the pen compared to the keyboard are smaller size and weight and a more natural style of writing for many people. Unfortunately, the version of the pen described in Section 2.1 produces too poor a position (acceleration) signal to allow successful recognition of characters. The new versions of the pen currently under development have the accelerometer replaced by the pressure sensors (though different from the current pressure sensor). The position signals obtained from

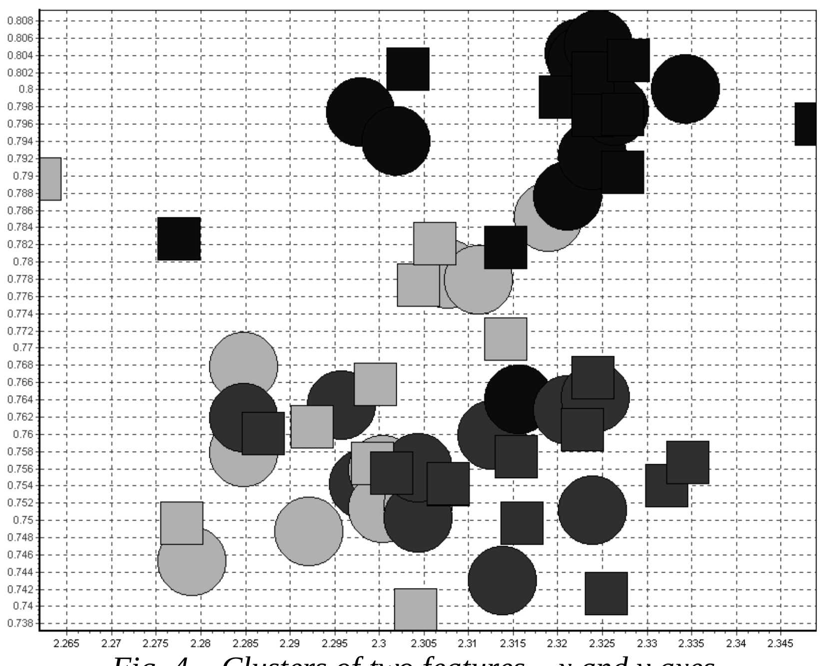

Fig. 4 – Clusters of two features – x and y axes
(Classes are distinguished by color)

the new versions of the pen are significantly better. The problem that we now have to face is how to integrate these sensors into a thin pen and how to place all the sensors around the cartridge so that they would not influence one another.

3 Methods

3.1 Signature verification using clustering

As usual, we do not apply recognition methods directly to raw data but we use preprocessing and feature extraction methods to eliminate a number of values representing an object. The features we use are based on statistical characteristics of signatures such as “maximum value of pressure signal” or “variance of acceleration signal amplitude”. Our recognizer uses a total of 20 features so far – each of them having a different weight in classification since some features are better than others. The features of one class create clusters in an n-dimensional space. To get an idea of how the clusters are distributed in space see graph (Fig. 4) where two features are shown – squares represent patterns; circles correspond to testing signatures

The main complication we encountered was the fact that signature verification differs from the general classification problem. The goal of the general classification problem is to choose one class from several classes, whereas the training data contain data from all classes. For our application all the training data are just patterns (authentic signatures). We have no training data for the second class – fake signatures.

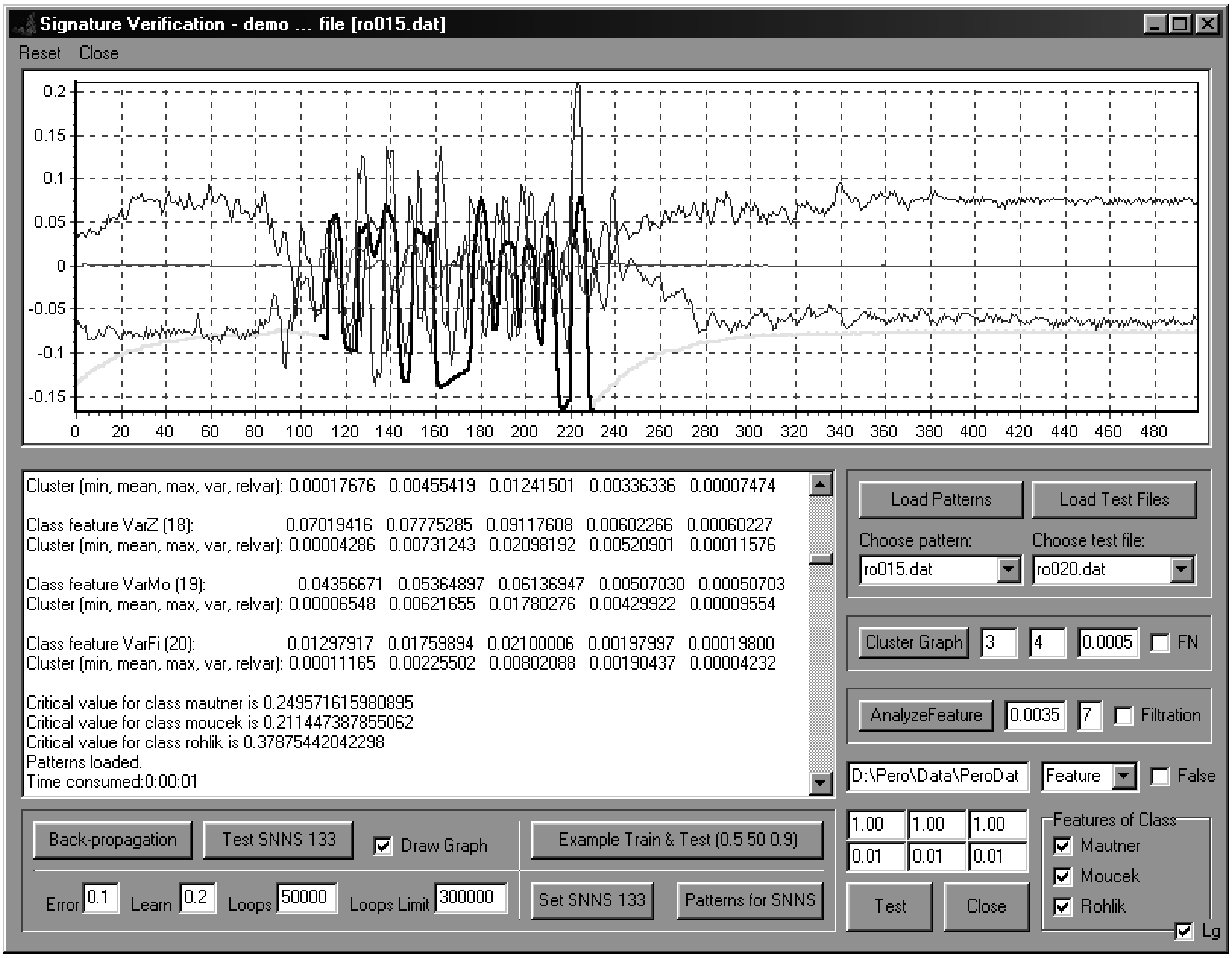

Fig. 5 – Program screenshot

In order to estimate what features are valuable and how accurate the developed methods are we implemented a complex recognition system (Fig. 5). The program allows us to analyze each signature separately, test several preprocessing methods and perform overall tests.

The basic idea of our method is straightforward – compute the distance between the tested signature and the pattern. If the distance is small then the tested signature is probably authentic. Now the problem is reduced to decision “What distance is small?”. We call that distance “critical cluster coefficient” and we compute it as a mean mutual difference between all pairs of patterns in the class. It means that the critical cluster value describes the similarity of signatures. For authors whose signatures are nearly the same this coefficient is low; in other words a signature is classified as authentic if it is very similar to some pattern. In contrast if the patterns are not uniform then the chance that the tested signature will be recognized is much greater thanks to the higher value of the critical cluster coefficient.

```
For each class C
    For each feature f
        For each pair of signatures Classes[C][i] and Classes[C][j]
            Compute the difference between Classes[C][i] and Classes[C][j] and add it
              to an extra variable Sum[f]
        Compute mean value mean[f] and variance var[f] of each feature over all pairs using
          the variable Sum[f]
    Compute critical cluster coefficient using variances var[f] and weights w[f] over all features f
```

Algorithm 1 – Signature verification training

```
For class C to be verified
        For each pattern Classes[c][i]
                For each feature f
                        Compute the difference and remember the least one over all patterns
        Sum up products of least differences and weights w[f] and compare the sum with
          Critical cluster coefficient
```

Algorithm 2 – Signature verification recognition

Although training is rather difficult results are not bad. Depending on the size of the training and testing data, the accuracy achieved is between 86 and 98 percent. As a correctly recognized signature we consider the situation when a fake is recognized as a fake or when an authentic signature is recognized as an authentic. A detailed description of experimental results is shown in Table 1.

Experiment number	1	2	3	4	5
Size of training data	30	30	60	30	30
Size of testing data	75	75	45	75	75
Number of fakes recognized as fakes	43	44	39	41	43
Number of originals recognized as oroginals	29	30	0	28	27
Numebr of fakes recognized as originals	2	1	6	4	2
Number of originals recognized as fakes	1	0	0	2	3
Overall accuracy ratio	72/75	74/75	39/45	69/75	70/75
Overall accuracy percentage	96%	98.7%	86.7%	92%	93.3%

Table 1 – Detailed description of experimental results

The data set used for training and testing can be considered fair because the authors who produced fakes were accurate and carefully trained to produce them. The data we used[2] can be found at URL `http://www.kiv.zcu.cz/~rohlik/pero`.

3.2 Author identification

Author identification is a problem slightly different from the signature verification although it looks similar at first sight. The new aspects are the following:

- Samples are classified into several classes – each class corresponds to one author,
- the written word is not a name (signature) but any other word – we use the same word for all authors.

Our approach is based on the idea that everyone writes in a specific way and therefore signals from different authors should differ. In general, graphologists use many signs to characterize the personality of the author. The four main sign categories are:

- movement (expansion in height and in width, coordination, speed, pressure, stroke, tension, directional trend, rhythm),
- form (style, letter shapes, loops, connective forms, rhythm),
- arrangement (patterns, rhythm, line alignment, word interspaces, zonal proportions, slant, margins – top, left and right),
- signature (convergence with text, emphasis on given name or family name, placement).

Unfortunately, all these signs cannot be used for classification – many of them require larger text samples than a single word. Our results show that four signs are sufficient for accurate classification: the mean and variance of the x signal and the mean and variance of the angle. These features correspond to expansion in height, coordination, speed, tension and rhythm.

Since we classify into several classes a neural net can be used as a classifier. After a few experiments we decided to use a two-layer perceptron network (Fig. 6). The features are fed into the input layer from the preprocessor by four channels. The internal part of the network contains 9 nodes (2N+1 rule). Three neurons in the output layer represent classes.

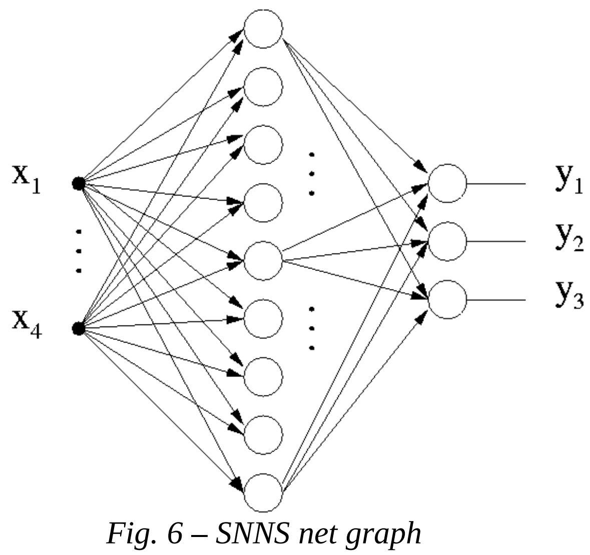

Fig. 6 – SNNS net graph

[2] Much more tests will be done in August 2001

The system is trained using a variant of the back-propagation algorithm with momentum. Results obtained from the testing data of 20 words per author were surprisingly good (see Table 2). The accuracy achieved by a well-trained network was close to 100%. Notice the accuracy drop caused by the increasing number of classes and decreasing number of words in the training set.

Experiment number	1	2	3	4	5	5
Number of classes (authors to be identified)	3	5	7	3	5	7
Size of training data	15	15	15	5	5	5
Size of testing data	5	5	5	15	15	15
Number of correctly classified words	15	24	33	39	71	101
Number of incorrectly classified words	0	1	2	6	4	4
Overall accuracy ratio	15/15	25/25	33/35	44/45	70/75	88/105
Overall accuracy percentage	100%	100%	94.28%	97.77%	93.33%	83.80%

Table 2 – Detailed description of experimental results

4 Conclusion and Future Work

This paper summarizes our experience with the new pen developed by our team. The purpose of our research is twofold – to improve the reliability of the signature verification and to make text recognition devices cheaper. Although results achieved so far are good, many more tests must be done in order to prove that our pen and methods are useful.

We have found that an acceleration sensor is not suitable for the text recognition. Therefore we plan to replace it by two pairs of pressure sensors that should produce better signals.

References

[1] J. Jeřábek: “Graphology – An Introduction to Graphological Diagnostics”, Tallpress, 1994.
[2] M. Pulver: “Symbolik der Handschrift”, Zürich, 1955.
[3] A. Zell et al: “SNNS – User Manual”, University of Stuttgart, 1995.
[4] R. P. Lippmann: “An Intorduction to Computing with Neural Nets”, IEEE ASSP Magazine April 1994, p.p. 4-22.
[5] S. Y. Kung: “Digital Neural Networks”, PTR Prentice-Hall, Inc., 1993.