Available online at www.sciencedirect.com
SCIENCE DIRECT®
Pattern Recognition Letters 26 (2005) 2400–2408
Pattern Recognition Letters
www.elsevier.com/locate/patrec

Identity authentication using improved online signature verification method

Alisher Kholmatov, Berrin Yanikoglu *

Sabanci University, Faculty of Engineering and Natural Sciences, Istanbul, 34956 Tuzla, Turkey

Received 24 April 2004; received in revised form 8 April 2005
Available online 28 June 2005

Communicated by T. Breuel

Abstract

We present a system for online handwritten signature verification, approaching the problem as a two-class pattern recognition problem. A test signature's authenticity is established by first aligning it with each reference signature for the claimed user, using dynamic time warping. The distances of the test signature to the nearest, farthest and template reference signatures are normalized by the corresponding mean values obtained from the reference set, to form a three-dimensional feature vector. This feature vector is then classified into one of the two classes (genuine or forgery). A linear classifier used in conjunction with the principal component analysis obtained a 1.4% error rate for a data set of 94 people and 619 test signatures (genuine signatures and skilled forgeries). Our method received the first place at SVC2004 with a 2.8% error rate.

Keywords: Online signature; Verification; Handwriting; Biometrics

1. Introduction

Biometric authentication is gaining popularity as a more trustable alternative to password-based security systems. Signature is a behavioral biometric: it is not based on the physical properties, such as fingerprint or face, of the individual, but behavioral ones. As such, one's signature may change over time and it is not nearly as unique or difficult to forge as iris patterns or fingerprints, however signature's widespread acceptance by the public, make it more suitable for certain lower-security authentication needs.

Signature verification is split into two according to the available data in the input. Offline (static)

* Corresponding author. Tel.: +90 216 483 9528; fax: +90 216 483 9550.
E-mail addresses: alisher@su.sabanciuniv.edu (A. Kholmatov), berrin@sabanciuniv.edu (B. Yanikoglu).

0167-8655/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2005.04.017

signature verification takes as input the image of a signature and is useful in automatic verification of signatures found on bank checks and documents. Online (dynamic) signature verification uses signatures that are captured by pressure-sensitive tablets that extract dynamic properties of a signature in addition to its shape. Dynamic features include the number and order of the strokes, the overall speed of the signature, the pen pressure at each point, etc. and make the signature more unique and more difficult to forge. As a result, online signature verification is more reliable than offline signature verification. Application areas of online signature verification include protection of small personal devices (e.g. PDA, laptop); authorization of computer users for accessing sensitive data or programs; and authentication of individuals for access to physical devices or buildings.

In an online signature verification system, the users are first enrolled by providing signature samples (reference signatures). Then, when a user presents a signature (test signature) claiming to be a particular individual, this test signature is compared with the reference signatures for that individual. If the dissimilarity is above a certain threshold, the user is rejected.

During verification, the test signature is compared to all the signatures in the reference set, resulting in several distance values. One then has to choose a method to combine these distance values into a single value representing the dissimilarity of the test signature to the reference set, and compare it to a threshold to make a decision. The single dissimilarity value can be obtained from the minimum, maximum or the average of all the distance values. Typically, a verification system chooses one of these and discards the others. Jain et al. (2002) report the lowest error rates with the minimum distance criterion.

Since obtaining actual forgeries is difficult, two forgery types have been defined: a *skilled* forgery is signed by a person who has had access to a genuine signature for practice. A *random* forgery is signed without having any information about the signature of the person whose signature is forged. In evaluating the performance of a signature verification system, there are two important factors: the false rejection rate (FRR) of genuine signatures and the false acceptance rate (FAR) of forgery signatures. As these two errors are inversely related, the equal error rate (EER) where FAR equals FRR is often reported. State-of-the-art results EER results for skilled forgeries are around 2.8% (Yeung et al., 2004).

2. Previous work

A comprehensive survey of signature verification can be found in (Leclerc and Plamondon, 1994; Plamondon and Lorette, 1989). Signature verification systems differ both in their feature selection and their decision methodologies. More than 40 different feature types have been used for signature verification (Jain et al., 2002; Ohishi et al., 2000; Vielhauer et al., 2002). Features can be classified in two types: global and local. Global features are features related to the signature as a whole, for instance the average signing speed, the signature bounding box, and Fourier descriptors of the signature's trajectory. Local features correspond to a specific sample point along the trajectory of the signature. Examples of local features include distance and curvature change between successive points on the signature trajectory. Most commonly used online signature acquisition devices are pressure sensitive tablets capable of measuring forces exerted at the pen-tip, in addition to the coordinates of the pen. The pressure information at each point along the signature trajectory is another example of commonly used local feature. In (Jain et al., 2002), some of these features are compared in order to find the more robust ones for signature verification purposes. Other systems have used genetic algorithms to find the most useful features (Yang et al., 1995).

Due to the high sampling rate of the tablet, some consecutive sample points may mark the same trajectory point, especially when the pen movement is slow. Most verification systems resample the input so as to obtain a trajectory consisting of equidistant points (Jain et al., 2002; Martens and Claesen, 1997; Yang et al., 1995). This is often done in order to remove redundant points to speed up the comparisons and to obtain a shape-based representation, removing the time

dependencies. Jain et al. (2002) separately keep track of the local velocity values and use them in aligning two signatures.

Due to the variability in signing speed, two signatures belonging to the same person may have different trajectory lengths (hence feature vectors of differing lengths). Therefore, the dynamic time warping algorithm with some variant of the Euclidian distance (Jain et al., 2002; Martens and Claesen, 1997; Ohishi et al., 2000; Parizeau and Plamondon, 1990) and Hidden Markov models (Van Oosterhout et al., 1998) are commonly used in aligning two signatures.

Number of signatures taken during the user enrollment also varies: between 3 and 20 samples are used in previous signature verification systems. The distance of the test signature to the closest reference signature has been found as the most useful (giving the lowest error rates) in (Jain et al., 2002), however other criteria, such as the average distance to the reference signatures or the distance to a template signature are also used. Template generation is generally accomplished by simply selecting one or more of the sample signatures as templates (Connell and Jain, 2001; Ohishi et al., 2000).

Various thresholds can be used in deciding whether the distance between the test signature and the reference and/or template signatures are acceptable. Two types of threshold selections are reported: writer dependent and writer independent thresholds (Jain et al., 2002). In the writer dependent scenario, thresholds are calculated for each user individually, whereas in the writer independent one, a global threshold for all the writers is set empirically during the training phase of the system.

State-of-the-art results for skilled and random forgeries are reported for several systems, however due to the differences in databases and forgery difficulties, most of them are not directly comparable. On the other hand, the First International Signature Verification Competition (SVC2004) organized in 2004 has tested more than 15 systems from industry and academia and the best results (around 2.6% EER) for skilled forgeries were obtained using the system presented in this paper (Yeung et al., 2004).

3. Proposed method

The proposed method shares the general approach and ideas suggested by some previous systems, but differs from the previous work in the selection of the features, the particular alignment algorithm and the decision mechanism using a classifier, taking as input normalized distances of the test signature from its claimed set, as explained in the following sections. The overall process is outlined below.

During the enrollment phase, the user supplies a set of reference (training) signatures which are used to determine user dependent parameters characterizing the variance within the reference signatures. The reference set of signatures, together with these parameters, are stored with a unique user identifier in the system's database.

When a test signature is input to the system for verification, it is compared to each of the reference signatures of the claimed person. The resulting minimum, maximum, and template distance values (shown in Fig. 2) normalized by the corresponding average values stored in the user's profile, form a three-dimensional feature vector used in classifying the test signature. These steps are explained in detailed in the following subsections.

3.1. Data acquisition

We have used Wacom's Graphire2 pressure sensitive tablet and pen. The tablet is capable of sampling data at 100 samples per second: at each sample point, the x, y coordinates of the signature's trajectory and the time stamp are recorded. Wacom's pen is featured to capture samples only during the interaction of the pen tip with the tablet.

3.2. Preprocessing

As mentioned before, most of the existing systems resample the signature in order to remove redundant points to speed up the comparisons and to obtain a shape-based representation, removing the time dependencies. However, resampling also results in significant loss of information since the seemingly redundant data incorporates

speed characteristics of the genuine signer. Another problem with resampling is that the critical points of the signature may be lost; critical points are sometimes added separately to the set of equidistant points obtained after resampling to solve this problem.

We found that as we initially expected, the benefits of not resampling significantly outweigh the disadvantage of not normalizing for speed, as found in our experiments (see Section 4).

3.3. Feature selection

For this system, we have experimented with the following three local features of the points on the signature trajectory: x–y coordinates relative to the first point of signature trajectory, the x and y coordinate differences between two consecutive points, and the curvature differences between two consecutive points. The x and y coordinate differences between two consecutive points (Δ_x, Δ_y) gave the lowest error rates and the results reported in Section 4 are with these features. Note that the Δ_x, Δ_y features are invariant with respect to translation. Also, in conjunction with the alignment algorithm explained below, they are quite successful in capturing similarities and ignoring irrelevant differences (e.g. a small length or angle changes between strokes cause small penalties).

Equal error rates with the other two features ranges between 6% and 11% according to the classifier used; these are reported in detail elsewhere (Kholmatov, 2003). In the rest of the paper, a signature in the system refers to a vector of dimension N, where N is the length of the signature and each vector element is 2D vector indicating the x–y offsets from the previous point.

We do not use dynamic features explicitly in the system. In particular, we have not found the pressure information to be useful and local or global speed is only *indirectly* taken into account during the signature alignment process (see Section 3.4).

We have done some limited studies to see the effects of using the pressure information as an additional local feature, but despite noticing a slight increase in the amount of pressure in forgeries, we have not found the pressure information to be useful in classification. The experiments with pressure were done for the First International Signature Verification Competition where there were two tasks: one where the pressure information was made available to the tested systems in one task and one without (Yeung et al., 2004). We have submitted the same system presented in this paper for both tasks, and even though we do not use any pressure information, our system had the lowest error in both tasks (with skilled forgeries). This suggests that the pressure information may not carry much useful information in terms of discriminating between forgeries and genuine signatures.

3.4. Signature alignment

In both training and verification stages, we need to align signatures to calculate their distances. In order to compare signatures of differing lengths, we have used the dynamic time warping (DTW) algorithm which is a widely used method for aligning vectors of different lengths (Bellman, 1957). Dynamic time warping algorithm finds the best non-linear alignment of two vectors such that the overall distance between them is minimized. The overall distance between two signatures S_1 and S_2 is calculated in linear time as shown in the following equation:

$$C[i,j] = \mathrm{Min}\begin{cases} C[i-1,j]+\gamma, \\ C[i,j-1]+\gamma, \\ C[i-1,j-1]+\mathrm{Dist}(S_1[i],S_2[j]), \end{cases} \tag{1}$$

where $S_i[j]$ denotes the i'th signature's j'th point on the trajectory and

$$\mathrm{Dist}(x,y) = \begin{cases} 0 & \text{if } ||x-y|| < \theta, \\ ||x-y|| - \theta & \text{otherwise.} \end{cases}$$

The above formulae show the well-known DTW algorithm, where C is the matrix to be filled by the algorithm and the γ is the constant coefficient penalizing a gap or an extraneous point in one of the signatures. The Dist function is designed to allow for insignificant variation in reference signatures by using the threshold constant θ. The result of applying the dynamic time warping algorithm $(C[\mathrm{length}(S_1), \mathrm{length}(S_2)])$ gives the dissimilarity score of two signatures.

Note that even though the dynamic time warping aligns signatures in differing lengths, the timing information that we intended to keep by not resampling is not lost: the speed difference between two signatures causes many points to be seen as extraneous points or missing points, adding γ to the match score at each extraneous point.

3.5. Enrollment

During enrollment to the system, the user supplies a number of signatures which are used in measuring the variation in a user's signatures, which is later used in the training and verification processes.

First, the supplied signatures are pairwise aligned to find the distances between each pair (Section 3.4). Using these alignment scores, we first select the reference signature with the minimum average distance to all other supplied signatures and designate it as the template signature. Then, statistics characterizing the spread/variation of the reference set signatures are calculated. Specifically, for a reference set R_{ID}, we compute average values over the reference set, for

- distances of reference signatures to their nearest neighbor ($\overline{d_{\min}}(R_{\mathrm{ID}})$),
- distances of reference signatures to their farthest neighbor ($\overline{d_{\max}}(R_{\mathrm{ID}})$),
- distances of reference signatures to the template signature ($\overline{d_{\mathrm{template}}}(R_{\mathrm{ID}})$).

The distances are similar to what is computed for a test signature, as illustrated in Fig. 2. The computed average distance values describe the user's variation to some extent and were selected so as to be used in normalizing a signature's min, max and template distances to its reference set. By normalizing these distances by the corresponding reference set averages, we eliminate the need to calculate user-dependent thresholds as explained in Section 3.6.

3.6. Training classifiers

A training data set consisting of 76 genuine signatures and 54 forgery signatures is collected in order to train classifiers used in the verification process. First, each training signature (Y) is compared to the signatures in the reference set (R_{ID}) it claimed to belong to, giving a three distance values ($d_{\min}(Y, R_{\mathrm{ID}})$, $d_{\max}(Y, R_{\mathrm{ID}})$, and $d_{\mathrm{template}}(Y, R_{\mathrm{ID}})$), as shown in Fig. 2. Note that the same normalization process is done both for training signatures during the training process, and for testing signatures during the testing process.

These distance values are then normalized by the corresponding averages of the reference set, to give the three-dimensional feature vector (F_Y):

$$F_Y = \begin{bmatrix} d_{\min}(Y, R_{\mathrm{ID}})/\overline{d_{\min}}(R_{\mathrm{ID}}) \\ d_{\max}(Y, R_{\mathrm{ID}})/\overline{d_{\max}}(R_{\mathrm{ID}}) \\ d_{\mathrm{template}}(Y, R_{\mathrm{ID}})/\overline{d_{\mathrm{template}}}(R_{\mathrm{ID}}) \end{bmatrix}.$$

The feature space corresponding to the training data is shown in Fig. 1 and supports that genuine (darker, blue dots)[1] and forgery (lighter, red stars) samples in the set are well separated with these normalized features. As seen in this figure, the genuine signatures are clustered around the origin while the forgeries show a wide spread, while the two classes are quite well separated. Hence, by training classifiers to find the boundary between genuine and forgery signatures, in terms of the deviations from the reference set means, we eliminate the need for user-dependent thresholds commonly used in deciding whether a signature is similar enough to the reference set.

In order to find the separating boundary between genuine and forgery signatures, we trained three classifiers: the Bayes classifier using the three-dimensional feature vectors assuming independent covariance matrices; a support vector machine (SVM) also using same feature vectors; and a linear classifier used in conjunction with the principal component analysis (PCA). Using PCA, we reduced the dimensionality from three to one while keeping most of the variance, as the three features are highly correlated. We then projected the data onto the principal component found with PCA and selected a threshold value separating the two classes. Note that we are modelling the genuine

[1] For interpretation of the references in colour, the reader is referred to the web version of this article.

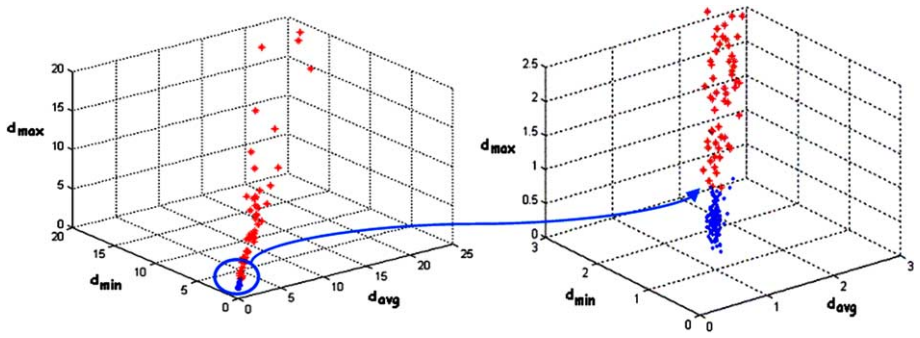

Fig. 1. Plot of genuine (blue dots) and forgery signatures (red stars) with respect to the three-dimensional distance vector. (For interpretation of the references in colour in this figure legend, the reader is referred to the web version of this article.)

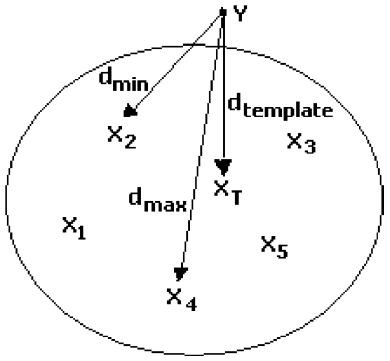

Fig. 2. The alignment distances used in the verification process: x_i represents ith reference signature supplied by the user and Y represents the signature to be verified. X_T is the template signature for which the overall distance to other reference signatures is minimum among all reference signatures. $d_{\max}$, $d_{\min}$, d_{template} represent distances to the furthest, nearest and template reference signatures, respectively.

and forgery classes of signatures, during the training phase. Once the training is complete, a new user is registered to the system *without* retraining the classifiers.

3.7. Verification

In order to verify a test signature, first the signature is compared to all the reference signatures belonging to the claimed ID. Then, the resulting distance values are normalized by the averages of the claimed reference set, as it was explained in Section 3.6 about training. The resulting three-dimensional feature vector is used in classifying the signature as genuine or forgery, using the classifiers trained before.

4. Results

The system performance was evaluated using the sample signatures supplied by 94 subjects enrolled to our system. Each subject supplied 10–15 genuine signatures for a total of 1134 signatures. There were no constraints on how to sign, nor was any information given about the working principles of the system, so that the subjects signed in their most natural way. Eight of the signatures given by each subject were used as the reference set for that user (total of 752 signatures), 76 genuine signatures in total were used for training the classifiers, and the rest (total of 306 signatures) was used in the evaluation of the system forming the first data set (DS1).

To collect skilled forgeries, we added a signing simulation module to our system. Simulation module animates the signing process of a given signature so that the forger could see not only the signature trajectory's points sequence but also the signing dynamics (speed and acceleration). Forgers had a chance of watching the signature's animation several times and practice tracing over the signature image a few times before forging it. Out of the 367 skilled forgeries obtained in this way, 54 were added to the training set and the remaining 313 signatures constituted the forgery data set (DS2). Table 1 summarizes the test data

Table 1
Data sets used for a performance evaluation of the system

Data set	Type	Size
Reference	Genuine	752
Training	Genuine/forgery	130
DS1	Genuine	306
DS2	Skilled forgeries	313

DS1 and DS2 are used to calculate FRR and FAR, respectively.

Table 2
Verification results obtained using Bayes, SVM, and the linear classifiers

Classifier	Without resampling		With resampling	
	FRR (%)	FAR (%)	FRR (%)	FAR (%)
Bayes	3.60	3.52	6.86	14.10
SVM	1.64	3.85	1.30	15.70
Linear	1.64	1.28	4.90	13.46

sets. Even though this is not a very large set, there is no public online signature database available.

Note that the training data composed of 76 genuine and 54 forgery signatures, is separate from both the reference set of genuine signatures and the test data used for performance evaluation. Fig. 3 shows sample signatures from our database showing some very easy and very difficult signatures to forge, highlighting the fact that signature is a biometric the complexity of which can be adjusted; this is useful since one can use different signatures for different security applications.

The results using both original and equi-distantly resampled signatures are shown in Table 2. For either of the signature types used (original or resampled) the results are with the Δ_x, Δ_y features, using the Bayes, SVM, and the linear classifiers, respectively. Best results were obtained using the linear classifier and not resampled signatures, with approximately a 1.4% total error rate (which is also roughly the equal error rate). The results of experiments with the Bayes classifier and the SVM were slightly inferior; this may be due to a poor fit to the assumed Gaussian distribution or the small number of training data used to estimate the model parameters. When used with resampled signatures, each of the classifiers performed significantly worse than when used with original signatures; this is due to the loss of important information incorporated in seemingly redundant sample points. Fig. 4 shows genuine copies and forgeries for a signature that are classified correctly by the system. Notice that even when the forgeries are visually very similar or the genuines appear different, the timing information may help with the discrimination.

Performance results obtained using only one of the distance values (distance to the closest, farthest, or template signature) are shown in Table 3. Although each criterion is effective in classifying either genuine or forgery signatures, none of them could successfully separate both genuine and forgery signatures.

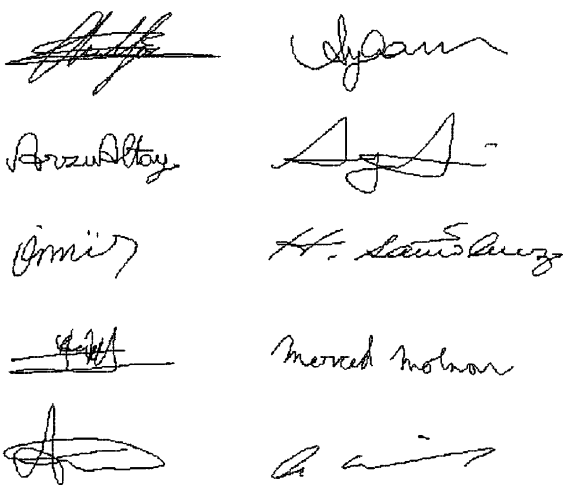

Fig. 3. Sample genuine signatures from the database. The top-most, left-most signature was very difficult to forge, while the right-most, bottom-most signature was quite easy.

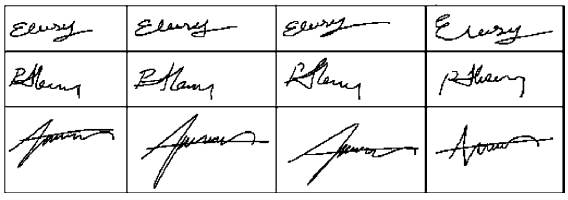

Fig. 4. Genuine copies (center) and forgeries (right) for sample signatures from the database (left). All four test signatures were correctly classified by the system.

Table 3
Verification results obtained using only one of the distance values with a threshold classifier

Classifier	Without resampling		With resampling	
	FRR (%)	FAR (%)	FRR (%)	FAR (%)
Minimum	3.27	1.92	4.90	12.78
Maximum	1.31	7.69	7.84	10.54
Template	4.90	1.92	9.48	11.82

Finally, to evaluate the system's performance against random forgeries, we used the reference signatures of each user as forgery signatures to all other users. Using the linear classifier and the data set of 69,936 ($94 \times 8 \times 93$) random forgeries constructed as described, we obtained a 1.06% error rate.

Even though these results are on a relatively small, proprietary database, the proposed system performed similarly at the First International Signature Verification Competition (Yeung et al., 2004), receiving the first place with the lowest average equal error rate (2.8%) when tested with skilled forgeries. In the part of the competition where pressure information was also available, our system again had the lowest error rate (2.9%) when tested with skilled forgeries. The second place systems had error rates of 4.4% and 5%, in these two categories, respectively.

5. Summary and conclusion

We have presented an online signature verification system with an improved decision criterion to separate genuine and forgery signatures. The main contribution of our system is to consider the signature verification problem as a two-class pattern recognition problem, using the various normalized distance values between a test signature and the claimed user's reference set, eliminating individual acceptance thresholds upon the enrollment of a new user. We have also studied the effects on the performance of some common approaches taken by previous signature verification systems and came up with improvements or suggestions for various steps of the process. For instance, we showed that with a suitable alignment algorithm, not resampling the signature is a direct way of preserving the very valuable timing information. Moreover, we showed that the pressure information does not seem to be a useful feature in distinguishing forgeries from genuines.

We experimented with three different classifiers and obtained a 1.4% overall error rate for a data set of 94 people and 619 signatures (genuine signatures and skilled forgeries). These results are quite good, given the fact that the forgeries used in the experiments were not random forgeries, as it is typically done, but relatively skilled forgeries. The system presented in this paper also received first place at the First International Signature Verification Competition (Yeung et al., 2004), with the lowest average equal error rate of 2.8% when tested with skilled forgeries.

References

Bellman, R., 1957. Dynamic Programming. Princeton University Press, Princeton, NJ.

Connell, S., Jain, A., 2001. Template-based online character recognition. Pattern Recognition 34 (1), 1–14.

Jain, A., Griess, F., Connell, S., 2002. On-line signature verification. Pattern Recognition 35 (12), 2963–2972.

Kholmatov, A., 2003. Biometric identity verification using online and off-line signature verification. Master's thesis, Sabanci University.

Leclerc, F., Plamondon, R., 1994. Automatic signature verification: The state of the art. Internat. J. Pattern Recognition Artificial Intelligence 8 (3), 643–660.

Martens, R., Claesen, L., 1997. Dynamic programming optimisation for on-line signature verification. In: ICDAR97. p. Poste.

Ohishi, T., Komiya, Y., Matsumoto, T., 2000. On-line signature verification using pen-position, pen-pressure and pen-inclination trajectories. In: ICPR, vol IV: pp. 547–550.

Parizeau, M., Plamondon, R., 1990. A comparative analysis of regional correlation, dynamic time warping and skeletal tree matching for signatures. IEEE Trans. Pattern Anal. Machine Intell. 12 (7), 710–717.

Plamondon, R., Lorette, G., 1989. Automatic signature verification and writer identification—state of the art. Pattern Recognition 22 (2), 107–131.

Van Oosterhout, J.J., Dolfing, H., Aarts, E., 1998. On-line signature verification with hidden Markov models. In: ICPR.

Vielhauer, C., Steinmetz, R., Mayerhofer, A., 2002. Biometric hash based on statistical features of on-line signatures. 16'th International Conference on Pattern Recognition.

Yang, X., Furuhashi, T., Obata, K., Uchikawa, Y., 1995. Constructing a high performance signature verification system using a ga method. In: 2nd New Zealand Two-Stream International Conference on Artificial Neural Networks and Expert Systems (ANNES '95), pp. 170–173.

Yeung, D., Chang, H., Xiong, Y., George, S., Kashi, R., Matsumoto, T., Rigoll, G., 2004. SVC2004: First international signature verification competition. In: Proc. Internat. Conf. on Biometric Authentication. pp. 16–22. Available from: http://www4.comp.polyu.edu.hk/~icba/.