Available online at www.sciencedirect.com
ScienceDirect
Engineering Applications of Artificial Intelligence 19 (2006) 693–704
Engineering Applications of ARTIFICIAL INTELLIGENCE
www.elsevier.com/locate/engappai

Support vector machines versus multi-layer perceptrons for efficient off-line signature recognition

E. Frias-Martinez*, A. Sanchez**, J. Velez

Biometry and Artificial Vision Group—GAVAB, Universidad Rey Juan Carlos, C/Tulipan s/n 28933 Móstoles, Madrid, Spain

Received 25 January 2005; received in revised form 21 October 2005; accepted 29 December 2005
Available online 10 March 2006

Abstract

The problem of automatic signature recognition has received little attention in comparison with the problem of signature verification despite its potential applications for accessing security-sensitive facilities and for processing certain legal and historical documents. This paper presents an efficient off-line human signature recognition system based on support vector machines (SVM) and compares its performance with a traditional classification technique, multi-layer perceptrons (MLP). In both cases we propose two approaches to the problem: (1) construct each feature vector using a set of global geometric and moment-based characteristics from each signature and (2) construct the feature vector using the bitmap of the corresponding signature. We also present a mechanism to capture the intrapersonal variability of each user using just one original signature. Our results empirically show that SVM, which achieves up to 71% correct recognition rate, outperforms MLP.

Keywords: Multi-layer perceptrons; Support vector machines; Off-line signature recognition

1. Introduction

Automatic human signature processing is a complex and specific area of automatic handwriting analysis (Han and Sethi, 1996; Madhvanath and Govindaraju, 2001; Plamondon and Shirari, 2000) with a high scientific and technical interest. There are two main research fields in this area: signature verification and signature recognition (or identification). The amount of interest and research efforts in these two fields is increasing due to the ability of human signatures to provide a secure process for authentication in many legal documents. The signature recognition problem consists on identifying the author of a signature. In this problem a signature database is searched to establish the identity of a given signer (Bajaj and Chaudhury, 1997; Lee and Pan, 1992). This task is different from signature verification. Verification defines the process in which a signature is tested to decide whether a particular signature truly belongs to a person (Bajaj and Chaudhury, 1997; Justino et al., 2003). The output in this case is either accepting the signature as valid or rejecting it as a forgery.

Automatic signature verification is an established and very active research field (Bolle et al., 2004; Leclerc and Plamondon, 1994; Plamondon and Lorette, 1989) with important applications to the validation of checks and other financial documents. Due to the demonstrated practical applications of signature verification, different techniques have already been applied: fuzzy logic (Ismail and Gad, 2000), geometric features (Fang et al., 1999; Hobby, 2000), global characteristics (Ramesh and Murty, 1999), genetic algorithms (Scholkopf et al., 1996), neural networks (Bajaj and Chaudhury, 1997; Baltzakis and Papamarkos, 2001; Velez et al., 2003), hidden Markov models (Camino et al., 1999), etc.In comparison, automatic signature recognition has received less attention, despite the potential applications that could use the signature as an identification tool (Pavlidis et al., 1994; Perez et al., 2004). For example, an automated signature recognition system could provide a company with a unique technique for validating the identity of each individual accessing to

*Corresponding author. Tel.: +34 916647452; fax: +34 914887049.
**Also for correspondence.
E-mail address: angel.sanchez@urjc.es (A. Sanchez).

0952-1976/$ - see front matter
doi:10.1016/j.engappai.2005.12.006

certain security-sensitive facilities (Lee and Pan, 1992). Other potential signature recognition applications are in law-enforcement applications, where the identification of perpetrators is a fundamental requirement of the solution, and in the analysis of some historical documents (Ismail and Gad, 2000). Some previous works in the area of automatic signature recognition are: Ammar et al. (1990) that uses a hierarchical scheme of signature descriptors to identify a test signature; (Han and Sethi, 1996), which considers a set of geometric and topologic features to map a signature image into two string of finite symbols; Pavlidis et al.(1998), which proposes the application of active deformable models for approximating the external shape of a signature; and Riba et al. (2000), that compares different statistical methods, using a feature extraction preprocessing, to carry out the recognition of signatures.

From a theoretical point of view, signature recognition and verification are different and independent problems, recognition is a 1:N matching problem while identification is 1:1. Apparently, the signature recognition problem looks more complex than the signature verification problem, and relatively little research effort has been focused on automatic signature recognition. In Ismail and Gad's work (2000), signature recognition and verification are treated as two separate and consecutive stages, where successful verification is highly dependent on successful recognition. Pavlidis et al. (1998) state that it would be of great value an intelligent signature identification system, which should be capable of arriving at a decision (recognition and verification) based only on the signature of the user. In this context, signature recognition is applied as an efficient preprocessing stage for signature verification. This approach and the potential applications of signature recognition, justify from our point of view the interest in finding effective automatic solutions for the recognition problem.

Signature recognition can be solved using off-line or on-line techniques. In the on-line approach the system uses not only the signature but also the data obtained during the signing process (dynamic information). The off-line approach only uses the digitalized image of a signature extracted from a document (static information).

In this paper we focus on the off-line signature recognition problem, which is the most common situation in many real applications (i.e. bank documents). A signature recognition system is characterized by two factors: (1) representation, which refers to the internal description that the system extracts from each signature of the data base and (2) match scheme, which involves the method that is used to select the best match from the set of identities of the signature data base. Han and Sethi (1996) use a similar terminology to describe the components of a signature recognition system.

Off-line signature recognition, and in general, image processing applications, face the problem of high dimensionality of the feature vectors. Because of that, a straightforward approach to the problem is to use pattern recognition techniques, like multi-layer perceptrons (MLP) with standard back-propagation learning or support vector machines (SVM), that have produced very good results in high dimensional classification problems (Cristianini and Shawe-Taylor, 2000; Lippmann, 1987).

Neural networks, in general, and MLP networks in particular, are widely used in handwritten recognition systems because they are very easy to train, very fast to use in classification decision process and generally achieve good performances in terms of correct recognition rate (Plamondon and Shirari, 2000). This popularity is related to the use of a back-propagation algorithm for the training process. The two main limits when using MLP in classification tasks are: (1) there is no theoretic relationship between the MLP structure and the classification task and (2) MLP derive hyperplanes separation surfaces, in feature representation space, which are not optimal in terms of margin between the examples of two different classes. Different neural networks architectures, including MLP, have already been used mainly for signature verification (Abbas, 1994; Bajaj and Chaudhury, 1997; Sethi and Han, 1995). The two main limitations that MLP face are solved by SVM: (1) by construction, SVM have a relationship between the structure (the support vectors) and the classification tasks and (2) SVM optimize the separation surfaces between two classes. SVM have been used very effectively for recognition applications like digit recognition (Gorgevik et al., 2001), face recognition (Guo and Chan, 2000) and 3D object recognition (Pontil and Verri, 1998). Bynm (2003) presents an extensive review of SVM pattern recognition applications. Recently they have been applied to on-line (Kholmatov, 2003) and off-line (Justino et al., 2003) signature verification problems. Nevertheless, to the best of our knowledge, SVM have not been applied to automatic human signature recognition.

SVM differ radically from MLP in that SVM training always finds a global minimum. The main difference between MLP and SVM is the principle of risk minimization. In case of SVM, structural risk minimization principle is applied by minimizing an upper bound on the expected risk whereas in MLP, traditional empirical risk minimization is used minimizing the error on the training data. The difference in risk minimization is to improve the generalization performance of SVM compared to MLP (Samanta et al., 2003).

This paper presents an off-line signature recognition system implemented with SVM as matching scheme and compares its performance with a more traditional MLP-solution in terms of correct classification rate. The approach to the problem, in both cases, is using two different representations: (1) using a feature vector constructed with global geometric and moment-based characteristics and (2) using the bitmap of the normalized image of each signature as the feature vector. The second approach is possible due to the ability of both SVM and MLP to work with high-dimensional problems. The paper also describes the construction of a human signature database using synthetic techniques. The main goal of this process is to create a representative training set which captures the intrapersonal variability of each writer without asking the user to provide his/her signature more that one time. The reason for that is caused by the fact that in many real applications it is not viable to get more than one signature from each system user for training purposes. The intrapersonal variability of the signature of each subject is captured by applying a combination of geometric transformations on the only original training signature. The process produces a relevant set of synthetic signatures for each writer that are used to train the two recognition systems. This is one of the main contributions of the paper, compared, e.g. with (Baltzakis and Papamarkos, 2001) where the training of the recognition system needs between 15–25 original signatures for each individual.

The rest of the paper is organized as follows. Section 2 presents an introduction to SVMs. Section 3 describes the basic characteristics of MLPs. Section 4 gives a state of the art of SVM and MLP in signature recognition. Section 5 justifies and describes the creation and preprocessing of the signature database used. Section 6 presents a signature recognition system constructed using SVM. Section 7 presents the proposed MLP-based signature recognition system and Section 8 compares the SVM and the MLP approach among themselves and with other references. Finally, in Section 9 conclusive remarks and future work are resumed.

2. Support vector machines

An SVM is a classifier derived from statistical learning theory first presented in (Boser et al., 1992). The main advantages of SVM when used for image classification problems are: (1) ability to work with high-dimensional data and (2) high generalization performance without the need to add a-priori knowledge, even when the dimension of the input space is very high. Excellent introductions to SVM can be found in (Cristianini and Shawe-Taylor, 2000; Vapnik, 1995).

The problem that SVMs try to solve is to find an optimal hyperplane that correctly classifies data points by separating the points of two classes as much as possible. Fig. 1 is an example of the previous situation. Given two classes, the objective is to identify the hyperplane that maximizes m:

$$m = \frac{2}{\|w\|} \tag{1}$$

while at the same time classifying correctly all the examples given. w^{T} being the hyperplane that verifies the previous condition, all points that are part of that class will verify: $w\, x+b>0$, where x is the point that is being validated. If a point is not part of that class, then: $w^{\mathrm{T}}x+b<0$. Formally the problem can be presented as follows. Let

$$(x_1, y_1), \ldots, (x_n, y_n) \in \Re^N \times Y, \quad y_i \in Y, \quad Y = \{-1, 1\} \tag{2}$$

be the set of labeled inputs, where -1 indicates that the input is not of that class and 1 indicates that the input is of that class. The decision boundary should verify

$$\begin{aligned} &x_i\, w^{\mathrm{T}} + b \geqslant 1, \quad \forall y_i = 1, \\ &x_i\, w^{\mathrm{T}} + b \leqslant -1, \quad \forall y_i = -1. \end{aligned} \tag{3}$$

The problem is solved by minimizing $\|w\|$ in order to maximize the margin m, subject to the conditions imposed by the training data:

$$\begin{aligned} &\text{maximize } m = \frac{2}{\|w\|} \\ &\text{subject to } y_i(x_i\, w^{\mathrm{T}} + b) \geqslant 1, \quad \forall i. \end{aligned} \tag{4}$$

Let $\alpha_1, \ldots, \alpha_N$ be the N non-negative Lagrangian multipliers associated with the constraints presented in Eq. (4). The problem of minimization is the equivalent to determining the saddle point of the function:

$$L_{\mathrm{p}}(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{N} \alpha_i\,(y_i(wx_i + b) - 1). \tag{5}$$

If we substitute the dual formulations of the constrains in L_{p}, the problem is transformed into

$$\begin{aligned} &\text{maximize } \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1, j=1}^{N} \alpha_i\, \alpha_j\, y_i\, y_j\, x_i^{\mathrm{T}}\, x_j \\ &\text{subject to } \alpha_i \geqslant 0, \quad \sum_{i=1}^{N} \alpha_i\, y_i = 0. \end{aligned} \tag{6}$$

This is a standard quadratic problem, where a global maximum α_i can always be found w can be recovered as

$$w = \sum_{i=1}^{N} \alpha_i\, y_i\, x_i. \tag{7}$$

Many of the α_i are zero, which implies that w is a linear combination of a small number of data. The set of elements x_i with non-zero α_i are called support vectors.

Graphically the support vectors are the set of points that mark the border of the class. This approach is valid whenever the set of points of the two classes are linearly separable. Nevertheless in real data this is usually not the case. In order to work with non-linear decision boundaries the key idea is to transform x_i to a higher dimension space (Fig. 2) using a transformation function Φ, so that in this new space the samples can be linearly divided. SVM solve

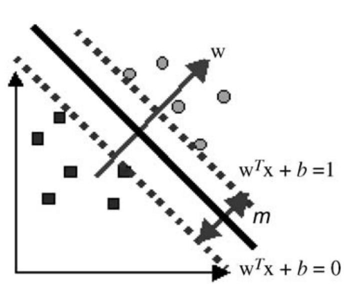

Fig. 1. Example of optimum hyperplane.

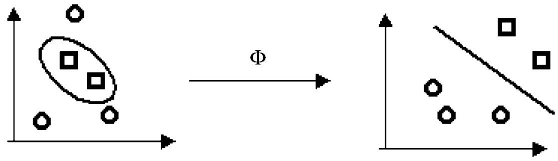

Fig. 2. Transformation of a non-linearly separable problem into a linearly separable problem.

these problems using kernels. The relationship between the kernel function K and Φ is

$$K(x_i, x_j) = \Phi(x_i)\,\Phi(x_j). \tag{8}$$

Intuitively, $K(x, y)$ represents the desired notion of similarity between data x and y. $K(x, y)$ needs to satisfy a technical condition (Mercer condition) in order for Φ to exist. An example of a kernel function is the Gaussian kernel, which is defined as

$$K(x_i, x_j) = \mathrm{e}^{-\|x_i - x_j\|^2/2\mathrm{std}^2}. \tag{9}$$

When working with a Gaussian kernel, std represents the standard deviation, and ideally should represent the minimum distance between any two elements of two different classes. As it can be seen when constructing a SVM based on a Gaussian kernel, the only value that needs to be defined is std. When working with kernels, in general it would not be possible to obtain w. Nevertheless SVM can be still be used. N_S being the number of support vectors of the training set, the decision function can be expressed as

$$f(x) = \sum_{i=1}^{N_S} \alpha_i y_i \Phi(x_i)\,\Phi(x) + b = \sum_{i=1}^{N_S} \alpha_i y_i K(x_i, x) + b. \tag{10}$$

Although the theoretical background given has introduced a classification system for only two classes, SVM can be generalized to a set of C classes. In this case each one of the classes will be trained against the rest C–1 classes, reducing the problem to a 2-class classification problem.

3. Multi-layer perceptron

MLPs are fully-connected feed-forward nets with one or more layers of nodes between the input and the output nodes. Each layer is composed of one or more artificial neurons in parallel. A neuron, as presented in Fig. 3, has N weighted inputs and a single output. A neuron combines these weighted inputs by forming their sum and, with reference to a threshold value and activation function, it will determine its output.

$x_1, x_2, \ldots, x_N$ being the input signals, $w_1, \ldots, w_N$ the synaptic weights, u the activation potential, θ the threshold and y the output signal and f the

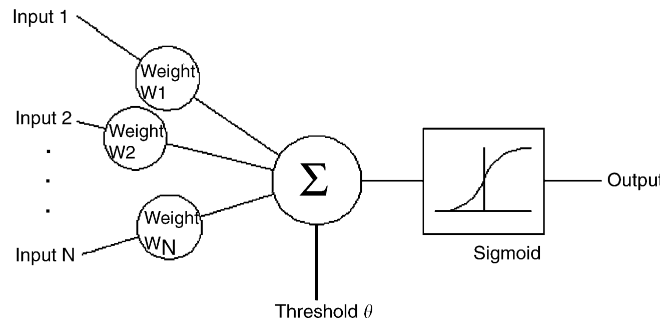

Fig. 3. Architecture of an artificial neuron.

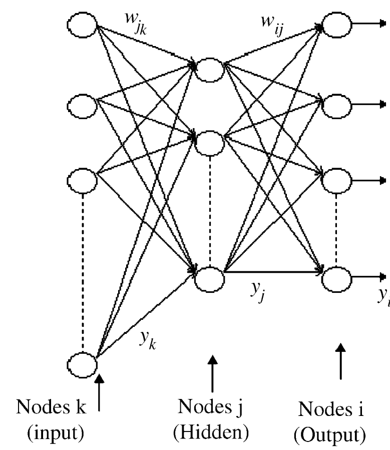

Fig. 4. Typical architecture of an MLP.

activation function:

$$u = \sum_{i=1}^{N} w_i x_i, \tag{11}$$

$$y = f(u - \theta). \tag{12}$$

Defining $w_0 = \theta$ and $x_0 = -1$, the output of the system can be reformulated as

$$y = f\left(\sum_{i=0}^{N} w_i x_i\right). \tag{13}$$

The activation function f defines the output of the neuron in terms of the activity level at its input. The most common form of activation function used is the sigmoid function.

Fig. 4 presents a two-layer perceptron with an input layer, one hidden layer and an output layer. Note that the input or branching nodes are not artificial neurons. Classification and recognition capabilities of MLP stem from the non-linearities used within the nodes. A single-layered perceptron implements a single hyperplane. A two-layer perceptron implements arbitrary convex regions consisting of intersection of hyperplanes. A three-layer perceptron implements decision surfaces of arbitrary complexity (Lippmann, 1987; Looney, 1997). That is the reason why a three-layer MLP is the most typical architecture.

MLP learn through an iterative process of adjustments applied to their free parameters. The most common learning algorithms are the standard back-propagation (Looney, 1997) and faster-learning variations (Fahlman, 1988). They use a gradient search technique to minimize a cost function equal to the mean square error (MSE) between the desired and the actual net outputs:

$$\mathrm{MSE} = \frac{1}{l}\sum_{i=1}^{l}(y_i - \hat{y}_i)^2. \tag{14}$$

The net is trained by initially selecting small random weights and internal thresholds, and presenting all training data repeatedly. Weights are adjusted after every trial using information specifying the correct class until weights converge and the cost function is reduced to an acceptable value. The generally good performance found for the back-propagation algorithm is somewhat surprising considering that it is a gradient descent technique that may find a local minimum in the cost function instead of the desired global minimum.

4. SVM and MLP for automatic off-line signature recognition

MLP and other neural networks architectures have mainly been used for signature verification systems (Abbas, 1994; Bajaj and Chaudhury, 1997; Baltzakis and Papamarkos, 2001). As previously pointed out, signature recognition systems have received little attention. Han and Sethi (1995), Han and Sethi (1996) and Sethi and Han (1995) present a signature recognition system that maps each signature into two strings of finite symbols obtained from the spatial distribution of geometric and topologic features. A local associative indexing scheme is then used to retrieve the identity of the signature. Different neural networks architectures have also been used for signature recognition systems. Baltzakis and Papamarkos (2001) present a two-stage perceptron classification structure for recognition and verification of human signatures. The first classifier combines the decision results of the neural network and the Euclidean distance obtained using three feature sets, and the second classifier uses a radial base function (RBF) neural network to take the final decision. Velez et al. (2003) present a signature recognition and verification system based on compression neural networks in combination with positional cuttings of the signature being tested. Other approaches to signature recognition systems, previously outlined in the Introduction Section, are Ammar et al. (1990); Han and Sethi (1996); Pavlidis et al. (1998); and Riba et al. (2000).

As far as we know SVM have not been used for signature recognition, but they have been used in other similar applications like handwritten digit recognition or recognition of some Asian characters. Gorgevik et al. (2001) use SVM for on-line digit recognition in combination with rule reasoning, and Kim (1998) use SVM for off-line recognition of handwritten Korean address strings. Bahlmann et al. (2002) use SVM for on-line handwriting recognition by designing a kernel able for sequential and non-fixed dimension data. Cun et al. (1995) compare different learning algorithms, including SVM, for handwritten digit recognition using the USPS (United States Postal Service) database of digits. SVM, using Gaussian kernels, can perform in this case as well as systems and algorithms designed specifically for this dataset, without including any detailed prior knowledge. Bellili et al. (2001) combine MLP with SVM for digit recognition. The proposed hybrid architecture is based on the idea that the correct digit class belongs to the two maximum outputs of the MLP, and that SVM can be introduced to detect the correct class among these two classification hypotheses.

5. Signature database: creation and preprocessing

There are inevitable variations in the signature patterns produced by the same person (intrapersonal variability) (Fang et al., 1999). One of the main problems of training a signature recognition system is to obtain a database of signatures extensive enough to capture all possible individual variations to allow the construction of a reliable system. The inexistence of referenced common signature benchmark databases and benchmarking rules makes very difficult the experimental systematic comparison of our method with other existing methods. For this reason, we have created our own database of signatures. Very recently, a first international competition aiming at objectively comparing different signature verification methods has just started (SVC, 2004).

Obtaining many signatures of each individual is a very tedious task, and, in general, it will be really difficult to obtain the collaboration of the system customers (e.g., in real environments like financial institutions).

In our system, each subject of the database was asked to sign just one time for training purposes. We considered that this is the only real approach to implement a signature recognition system for real and practical applications. A user of a bank, e.g., will be willing to give one signature for the bank account, but no more that that. For testing purposes we also asked for five more original signatures of each user, thus having a total of six original signatures per user. The process was done using different pens and with no restrictions. The original database is composed of 38 individuals with a total of 228 original signatures. Signatures were scanned into binary images with a 200 dpi resolution and stored in BMP format.

One of the main goals of our approach is to be able to create an efficient signature recognition system using just one original signature for training purposes. Nevertheless, with only one original signature the intrapersonal variability is not captured. In order to have enough data to construct an efficient classifier, the original signature of

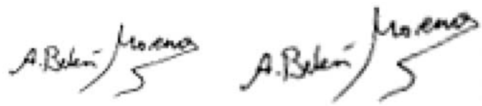

Fig. 5. Example of +5 rotation and +10 scaling (right) of the original signature (left).

each individual is used to obtain 300 synthetically generated signatures. Using as seed the original signature, a set of transformations are applied that mimic the intrapersonal variability. The transformations applied are: rotations in the range [15°,15°], scalings in the range [−20%, 20%], horizontal and vertical displacements in [−20%, 20%] and different types of noise additions (e.g. adding random black pixels). Fig. 5 presents an example of a +5° rotation combined with a +10% scaling of the original signature. These transformations were applied in different combinations for each one of the original signatures. Each one of those signatures was then normalized (using a bilinear interpolation algorithm) to a rectangle measuring 48×24 pixels (total of 1152 pixels). The size of the normalized signatures was obtained as the average value of the size of the signatures. Running the same process for each original signature produced a set of 11,400 signatures, with 300 signatures for each individual. The database containing the original and the synthetic signatures used for the experiments described in the following sections can be found in: http://gavab.escet.urjc.es.

This set of 11,400 signatures is used for training the signature recognition system. The other five original signatures per user, which make a total of 190 signatures, constitute the testing set. Note that the training set has been generated with only one original signature and that the testing set is linearly independent from the training test (there are no synthetic signatures in the testing set).

Fig. 6 presents the architecture of the recognition system when constructed using SVM. The corresponding figure for the proposed MLP-based recognition system is equivalent if MLP instead SVM is used. Signature images are basically a collection of points distributed over a well-defined area, this means that a numerical representation can be obtained. From the testing and the training set, two representations of each signature are obtained, one using the bitmap of the image, and another one that uses a set of characteristics or features of the signature. Some approaches for off-line signature verification use global geometric and moment-based features of the signatures (Bajaj and Chaudhury, 1997; Looney, 1997). This classification of the features is based on how they are obtained. The next two subsections describe the preprocessing of each signature in order to obtain these two sets of signature characteristics.

5.1. Global geometric characteristics

In this subsection, we introduce the concepts of center of gravity, horizontal and vertical base-line, least-square line and some shape measures.

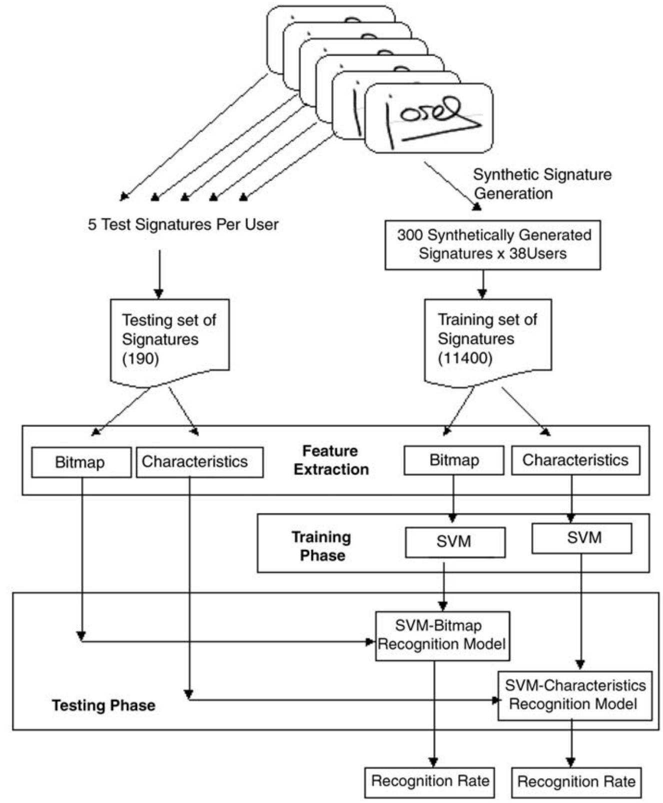

Fig. 6. Architecture of the proposed SVM-based signature recognition system.

Fig. 7. Example of signature projection in the X and Y axes.

Horizontal (P_H) and vertical (P_V) projection images reflect the distribution of signature pixels along X and Y directions. The projections of the signature are computed as the sum of the black pixels of the image in each row or column. Fig. 7 shows an example of the projection of a signature in both X and Y axes. These projection images can, respectively, be defined in the following way:

$$P_H[y] = \sum_{x=1}^{x_{max}} b[x,y], \quad \text{for } y = 1,2,\ldots,y_{max}, \tag{15}$$

$$P_V[x] = \sum_{y=1}^{y_{max}} b[x,y], \quad \text{for } x = 1,2,\ldots,x_{max}, \tag{16}$$

where $b[x,y] \in \{0,1\}$ indicates the pixel at the xth row and yth column.

The horizontal (C_H) and vertical (C_V) centers of gravity of a signature are computed from the projection images as

$$C_H = \frac{\sum_{x=1}^{x_{max}} x P_V[x]}{\sum_{y=1}^{x_{max}} P_V[x]}, \tag{17}$$

$$C_V = \frac{\sum_{y=1}^{y_{max}} y P_H[y]}{\sum_{y=1}^{x_{max}} P_H[y]}. \tag{18}$$

Also, the signature is divided in four cells, and for each one of those cells the corresponding vertical and horizontal coordinates of the center of gravity are obtained.

The horizontal and vertical baselines are obtained from the horizontal and vertical projections of the signature. Formally, the horizontal B_H and vertical B_V baselines are defined as

$$B_H = \max\{P_H[y]\}, \quad \text{for } x = 1, \ldots, x_{max}, \tag{19}$$

$$B_V = \max\{P_V[x]\}, \quad \text{for } y = 1, \ldots, y_{max}. \tag{20}$$

The least-square line is defined by the parameters b and m as

$$y = mx + b. \tag{21}$$

Fig. 8 presents an example of the least-square line of an example signature. N being the number of black pixels of the signature, and (x_1,y_1), (x_2,y_2), …, (x_N,y_N) the coordinates of each one of those pixels, the least-square line describes the trend of the signature pixel set. The corresponding parameters b and m are computed as

$$b = \frac{(\sum_{i=1}^{N} x_i^2)(\sum_{i=1}^{N} y_i) - (\sum_{i=1}^{N} x_i)(\sum_{i=1}^{N} x_i y_i)}{N(\sum_{i=1}^{N} x_i^2) - (\sum_{i=1}^{N} x_i)^2}, \tag{22}$$

$$m = \frac{N(\sum_{i=1}^{N} x_i y_i) - (\sum_{i=1}^{N} x_i)(\sum_{i=1}^{N} y_i)}{N(\sum_{i=1}^{N} x_i^2) - (\sum_{i=1}^{N} x_i)^2}. \tag{23}$$

Shape measures are physical dimensional values that characterize the appearance of an image signature. These characteristics include, among others: area, perimeter, area/perimeter ratio, 4 and 8-connected components, roundness, compactness, area of the convex hull, maximum axis of the convex hull and angle of the maximum axis.

Area and perimeter are computed using the bounding box of the signature (Fig. 9). In order to obtain this boundary measure, first the superior horizontal limit (SH), superior vertical limit (SV), inferior horizontal limit (IH) and inferior vertical limit (IV) of the signature need to be extracted. This can be done by scanning each line and each row of the image of the signature until the first and last black pixels are found. The bounding box of a signature is the smallest rectangle that contains that signature. Being $l_1 = \text{SV–IV}$ and $l_2 = \text{SH–IH}$ the size of the rectangle, the area of the signature A, the perimeter of the signature P and the area/perimeter ratio R, are defined as

$$A = l_1 \, l_2, \tag{24}$$

$$P = 2l_1 + 2l_2, \tag{25}$$

$$R = \frac{A}{P}, \tag{26}$$

The concept of connectivity of two pixels indicates if those two pixels are part of the same object. Two pixels are connected if they are adjacent. Two pixels p and q are 4-connected if q is part of $N4(p)$, where $N4(p)$ represents the 4-neighbours of p. Fig. 10 presents the 4-neighbours of pixel $p = (x, y)$. The concept of 8-conectivity is analogously defined using the set $N8(p)$ for each image pixel p.

The convex hull of a signature captures the signature shape and is defined as the smallest convex set containing the black pixels of the signature. Fig. 11 presents an example of the convex hull of a signature. Once the convex hull of the signature has been obtained, the area of the convex hull, area_CH, can be obtained. The main axis of the convex hull A (Fig. 11), is defined as the biggest

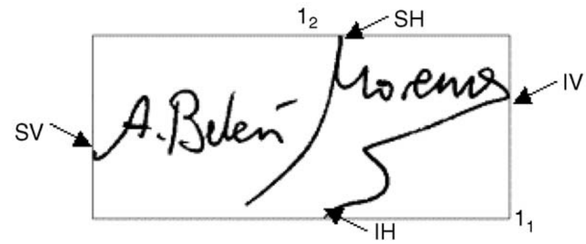

Fig. 9. Example of the bounding box that contains the signature.

	(x, y-1)	
(x-1,y)	(x, y)	(x+1,y)
	(x,y+1)	

Fig. 10. 4-Neighbours of pixel $p = (x,y)$.

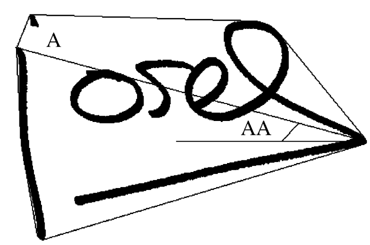

Fig. 11. Convex hull and main axis of a signature.

Fig. 8. Example of least-square line of a signature.

distance between any two points of the hull. The angle of the main axis AA (Fig. 11), is defined as the angle between the main axis and the horizontal. Finally, roundness R and compactness C parameters are defined as

$$R = \frac{\text{area\_CH}}{A^2}, \tag{27}$$

$$C = \frac{\sqrt{\text{area\_CH}}}{A}. \tag{28}$$

5.2. Moment-based characteristics

These set of measures are defined using the vertical and horizontal projections, introduced in the previous subsection, and the concepts of projection moments. Formally an r-order horizontal moment is defined as

$$\mu_r^{\mathrm{H}} = \sum_{x=1}^{x_{\max}} (x - x^{\mathrm{c}})^r \, P_{\mathrm{H}}(x), \tag{29}$$

where x^{c} is the center of the projection of the signature. The concept of r-order vertical moment is equivalently defined for the Y axis.

The moment-based characteristics include, among others: kurtosis, skewness and the relative projection coefficients. The vertical and horizontal kurtosis in a signature indicates how the histogram of the projection is divided among the central part and the inferior and superior bounds. In other words, it shows the importance of the center of the image. Formally the vertical and horizontal kurtosis measures are defined as

$$K_{\mathrm{H}} = \frac{\mu_4^{\mathrm{H}}}{(\mu_2^{\mathrm{H}})^2}, \tag{30}$$

$$K_{\mathrm{V}} = \frac{\mu_4^{\mathrm{V}}}{(\mu_2^{\mathrm{V}})^2}. \tag{31}$$

The skewness indicates the factor of asymmetry in the distribution of the projection moments. Horizontal and vertical skewness are defined as

$$S_{\mathrm{H}} = \frac{\mu_3^{\mathrm{H}}}{(\mu_2^{\mathrm{H}})^{1.5}}, \tag{32}$$

$$S_{\mathrm{H}} = \frac{\mu_3^{\mathrm{H}}}{(\mu_2^{\mathrm{H}})^{1.5}}. \tag{33}$$

Finally, we also use as features some moment relations because they are relatively insensitive to distortions and style variations. Two considered measures are the relative horizontal and vertical projection coefficients (Baltzakis and Papamarkos, 2001), which are respectively computed as

$$\mathrm{VH}_1 = \frac{\mu_2^{\mathrm{V}}}{(\mu_2^{\mathrm{H}})}, \tag{34}$$

$$\mathrm{VH}_2 = \frac{\mu_4^{\mathrm{V}}}{(\mu_4^{\mathrm{H}})}. \tag{35}$$

6. Off-Line SVM signature recognition

This section describes the construction of a signature recognition system using SVM. The package used for training the SVM-based signature recognition system was SVMTorch (Collobert and Bengio, 2002; Collobert et al., 2001). A Gaussian kernel was used to run all the experiments. The reason for choosing this kernel is that it has been widely used with very good results for pattern recognition applications (Scholkopf et al., 1996). Although not detailed in this paper, other tests where run using lineal and polynomial kernels obtaining poorer results.

Two experiments were run in order to test the efficiency of SVM to recognize human signatures. The first experiment constructed a signature recognition system using as feature vector the set of global geometric and moment-based characteristics. Each signature was represented in this case by a 37-dimensional vector containing the previously defined characteristics. In the second experiment, due to the ability of SVM to work with high dimensional data, each signature was represented using as feature vector its 24×48 bitmap normalized in the range [0,1], which produced a 1152-dimension vector.

Figs. 12 and 13 present for both experiments the results of testing the set of 190 original signatures with the SVM trained with the set of synthetically generated signatures.

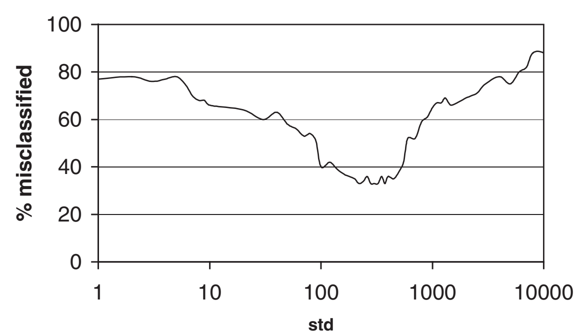

Fig. 12. Percentage of misclassified signatures when using SVM trained with characteristic feature vectors.

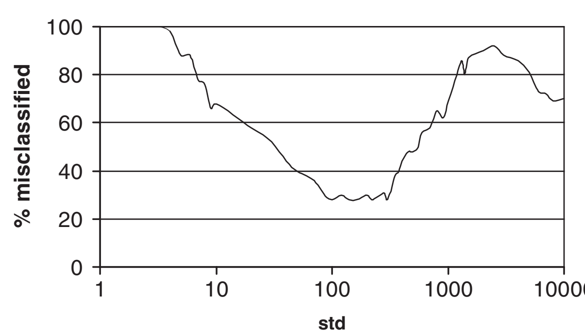

Fig. 13. Percentage of misclassified signatures when using SVM trained with bitmap feature vectors.

The X-axis presents, in a logarithmic scale, the values of the standard deviation (std) of the Gaussian kernel. The Y-axis presents the percentage of incorrect classified signatures. Table 1 summarizes for both experiments the optimum range for std and the respective percentages of incorrectly classified signatures.

Bitmaps produced a better recognition rate that characteristics. As expected the more information the feature vectors used to construct the recognition system the better the results were. These results also show that, in combination with SVM, the synthetically generated database captures the intrapersonal variability of a signature to an acceptable extent. These results are very promising to further study the process of synthetically generating the training data using just one original signature as seed.

The main problem of using bitmaps as feature vectors is the time needed both to train and to test the system due to the high dimensionality of the vectors. In a SVM the training and response time depends mainly on the dimension of the vectors, more that on the number of testing vectors. Comparing the training time of the bitmap approach with the characteristic approach for the range of std where the optimum values are obtained, the time needed is approximately 15 times bigger. In the testing process the response time of the characteristic approach for one signature is on average 0.007 s compared with the 0.11 s response time of the bitmap approach.

Table 1
Summarization of SVM experiments

	std	% Misclassified
Characteristics	[220,460]	33.5
Bitmap	[70,300]	28.8

7. Off-Line MLP signature recognition

This section describes the implemented MLP-based signature recognition system and presents its results. MLP are a traditional pattern recognition approach, thus the interest in comparing its results with the SVM approach. As for SVM, two experiments were run, one using as feature vector the set of global geometric and moment-based characteristics and the other one using the bitmap image of each signature. The characteristics vector was normalized in the range [−1,1] and the bitmap vector in [0,1].

The tool used for both experiments was JavaNNS, (Java Neural Network Simulator, 2003). In both experiments a fully connected MLP with backpropagation learning algorithm was implemented. When using the characteristics feature vector the network had 37 inputs (one for each characteristic) and 38 outputs (one per user). For the bitmap MLP-architecture the network had 1152 inputs and 38 outputs. In both experiments the learning process used an initial random generation of weights in the range [−1,1], standard backpropagation algorithm and a learning rate (step size) of $\rho = 0.001$ for the characteristics vector and $\rho = 0.1$ for the bitmap feature vector. In both architectures the network had two hidden layers. When using the characteristics feature vector the first internal layer had 10 neurons and the second, 20. For the bitmap architecture the first hidden layer had 80 neurons and the second, 20. Although we tested other number of neurons and other architectures the previous configurations provided the best classification rates. Fig. 14 presents a part of the training evolution for the bitmap (which was executed for 1000 iterations) and for the characteristics vector (which was executed for 10000 iterations). Fig. 15 represents respectively the MSE error corresponding to each one of the 190 used test patterns for the bitmap and characteristics

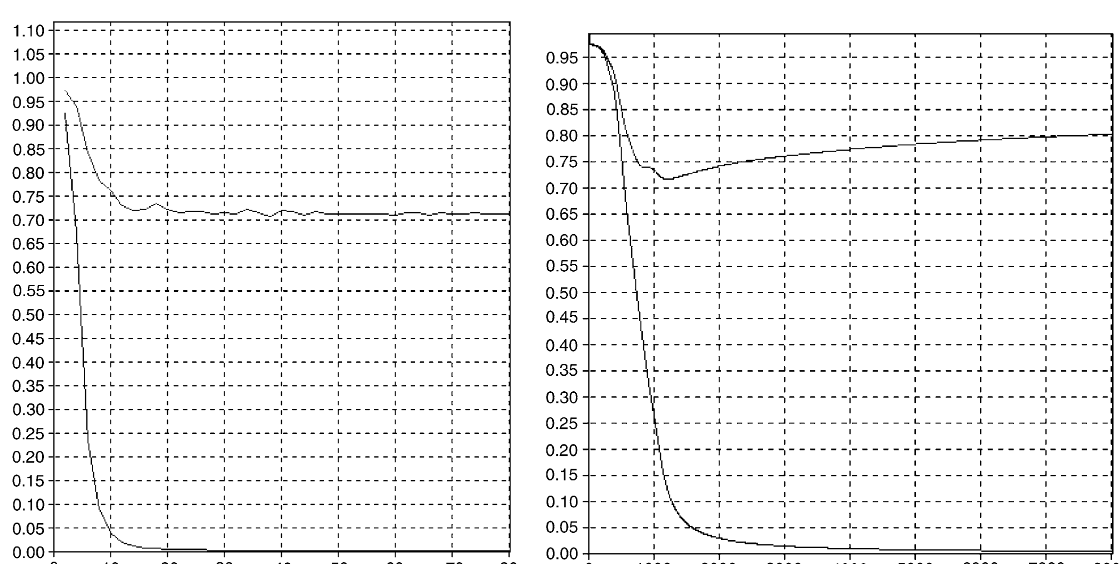

Fig. 14. MLP-training evolution using as feature vector the bitmap (left) and the characteristics (right).

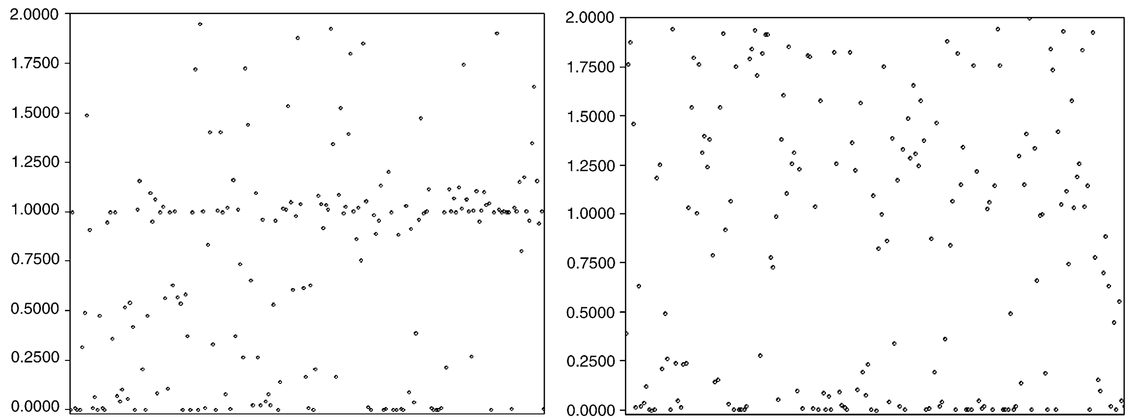

Fig. 15. MSE error for each signature when using as feature vector the bitmap (left) and the characteristics (right).

vectors. Both approaches produced a very similar classification rate, with the characteristics feature vector providing a 45.2% correct classification rate and the bitmap feature vector a 46.8% correct classification rate.

8. Analysis of the experimental results

Table 2 compares the correct recognition rate obtained by SVM and MLP using our signature database. As it can be seen, SVM outperforms MLP, for the process of signature recognition, with both approaches. Also, SVM training time was much shorter that MLP, nevertheless for this comparison, the difference in code efficiency of the tools used should be taken into account.

The bitmap approach, both for SVM and MLP, produced better results than the characteristics feature vector. Nevertheless, also for SVM and MLP, the training time needed when using the characteristics vector was between 7 and 12 times smaller than the corresponding to the bitmap approach.

We have found in the literature some references that to some extent, can be compared with our approach. Bajaj and Chaudhury, 1997 reported an 80.1% correct classification rate for signatures using neural networks and a set of features similar to the one we have presented. Nevertheless, in this case the authors use around 15–25 originals signatures for each user. We consider that although the results are superior, the approach to the problem is not realistic and cannot be used to implement practical signature recognition applications.

Pavlidis et al., 1994 reported 70.8% correct classification rates of signatures, compared to our 71.2% using SVM, in a very similar environment, using just one original signature and a similar number of users. Nevertheless in this case the authors have developed a complex ad-hoc solution to the problem (revolving deformable models) while our approach is much more standard and simple because it is based on a well-known classification mechanism like SVM and MLP.

Table 2
Percentage of correct recognition rates for SVM and MLP classifiers

	SVM (%)	MLP (%)
Characteristics	66.5	45.2
Bitmap	71.2	46.8

9. Conclusions

Despite the potential applications for accessing security-sensitive facilities and for processing legal and historical documents of the signature recognition problem, it has received very little attention when compared with signature verification. The relevance of signature recognition, apart from its applications, also comes from the fact that it can be considered a fundamental preprocessing stage for signature verification. In other words, a correct signature verification depends on a correct signature recognition.

This paper has presented an efficient off-line signature recognition system constructed using SVM. To the best of our knowledge this is the first application of SVM for signature recognition. MLP and SVM have been applied to many classification problems, generally yielding good performance. In this paper we also compare these two machine-learning algorithms on signature recognition. From the results obtained we empirically conclude that SVM work better than MLP, with standard backpropagation learning, for off-line signature recognition (within our signature database), both for the identification rate obtained (there is an increment of 20% in the recognition rate when using SVM) and for the training time needed. This superior SVM performance is due to the superior generalization ability of support vector machines in high-dimensional spaces.

We have also presented an original technique for the synthetic generation of a training signature database, thus avoiding the inconvenience that each user has to sign more that one time in order to be correctly recognized by the system. The system generates a significant set of synthetic signatures for training purposes using just one original signature as seed. This is a very important characteristic in order to create practical applications, because in a real environment a user will be willing to provide one original signature, but in general, no more than that. The recognition results presented in the paper prove that, when combined with SVM, this mechanism is able to capture, to a large extent, certain possible classes of intrapersonal variability.

Regarding future research lines, first we plan to develop a more complete off-line signature database freely available to the research community in order to provide the means to compare different signature recognition techniques. With respect to the signature recognition system, regarding the good results provided by the synthetic generation of signatures, we plan to introduce also nonlinear modifications in order to optimize the capture of intrapersonal variability. Another future research line is to investigate on the design of an ad-hoc kernel for the process of signature identification in order to improve the efficiency of the SVM approach.

Acknowledgement

This work has been partially supported by the Direccion General de Universidades e Investigacion of the Comunidad Autonoma de Madrid (Spain) through its postdoctoral research program.

References

Abbas, R., 1994. A prototype system for off-line signature verification using multilayered feedforward neural networks. Master Thesis, Royal Melbourne Institute of Technology (RMIT), Department of Computer Science, Melbourne, Australia.

Ammar, M., Yoshida, Y., Fukumura, T., 1990. Structural description and classification of signature images. Pattern Recognition 23, 697–710.

Bahlmann, C., Haasdonk, B., Burkhardt, 2002. On-line handwriting recognition using support vector machines—a kernel approach. Proceedings of International Workshop on Frontiers in Handwriting Recognition (IWFHR). pp. 49–54.

Bajaj, R., Chaudhury, S., 1997. Signature verification using multiple neural classifiers. Pattern Recognition 30, 1–7.

Baltzakis, H., Papamarkos, N., 2001. A new signature verification technique based on a two-stage neural network classifier. Engineering Applications of Artificial Intelligence 14, 95–103.

Bellili, A., Giloux, M., Gallinari, P., 2001. An hybrid MLP–SVM handwritten digit recognizer. Proceedings of Sixth International Conference on Document Analysis and Recognition (ICDAR'01). pp. 28–33.

Bolle, R.M., Connell, J.H., Pankanti, S., Ratha, N.K., Senior, A.W., 2004. Guide to Biometrics. Springer, New York.

Boser, B. E., Guyon, I. M., Vapnik, V. N., 1992. A training algorithm for optimal margin classifiers. Proceedings of Fifth Annual Workshop on Computational Learning Theory. pp. 144–152.

Bynm, H., 2003. A survey on pattern recognition applications of Support Vector Machines. International Journal of Pattern Recognition and Artificial Intelligence 17, 459–486.

Camino, J.L., Travieso, M.C., Morales, C.R., Ferrer, M.A., 1999. Signature classification by hidden Markov model. Proceedings of IEEE International Carnahan Conference on Security Technology. pp. 481–484.

Collobert, R., Bengio, S., 2002. SVMTorch: support vector machines for large-scale regression problems. Journal of Machine Learning Research 1, 143–160.

Collobert, R., Bengio, S., Mariéthoz, J., 2001. Torch: a modular machine learning software library. Technical Report IDIAP-RR 02-46, IDIAP.

Cristianini, N., Shawe-Taylor, J., 2000. An Introduction to Support Vector Machines. Cambridge University Press, Cambridge.

Fahlman, S.E., 1988. Faster-learning variations on back-propagation: An empirical study. In: Sejnowski, T.J., Hinton, G.E., Touretzky, D.S. (Eds.), Proceedings of 1988 Connectionist Models Summer School. Morgan Kaufmann, San Mateo, CA.

Fang, B., Wang, Y.Y., Leung, C.H., Tang, Y.Y., Kwok, P.C.K., Tse, K.W., Wong, Y.K., 1999. A smoothness index based approach for off-line signature verification. Proceedings of ICDAR'99, pp. 785–787.

Gorgevik, D., Cakmakov, D., Radevski, V., 2001. Handwritten digit recognition by combining support vector machines using rule-based reasoning. In: Proceedings of the 23rd International Conference Information Technology Interfaces (ITI). pp. 139–144.

Guo, G., Li, S. Z., Chan, K., 2000. Face recognition by support vector machines. In: Proceedings of Fourth IEEE International Conference on Face and Gesture Recognition. pp. 196–201.

Han, K., Sethi, K., 1995. Signature identification via local association of features. Proceedings of Third International. Conference on Document Analysis and Recognition, Montreal, Canada. pp. 187–190.

Han, K., Sethi, K., 1996. Handwritten signature identification. Pattern Recognition Letters 17, 83–90.

Hobby, J.D., 2000. Using shape and layout information to find signatures, text, and graphics. Computer Vision and Image Understanding, 80, 88–110.

Ismail, M.A., Gad, S., 2000. Off-line Arabic signature recognition and verification. Pattern Recognition 33, 1727–1740.

Java Neural Network Simulator, 2003. http://www-ra.informatik.uni-tuebingen.de/software/JavaNNS/welcome.html.

Justino, E.J.R., Bortolozzi, F., Sabourin, R., 2003. An off-line signature verification method based on SVM classifier and graphometric features. Internationa. Conference on Advances in Pattern Recognition (ICAPR 2003), Calcutta, India. pp. 10–13.

Kholmatov, A.A., 2003. Biometric identity verification using on-line and off-line signature verification. M.Sc. Thesis, Sabanci University, Istanbul, Turkey.

Kim, S.H., 1998. Recognition of handwritten Korean address strings by tight-coupling of minimum distance classification and dictionary-based post-processing. Computer Processing of Oriental Languages 12, 207–221.

Le Cun, Y., Jackel, L.D., Bottou, L., Brunot, A., Cortes, C., Denker, J.S., Drucker, H., Guyon, I., Muller, U.A., Sackinger, E., Simard, P., Vapnick, V., 1995. Comparisson of learning algorithms for handwritten digit recognition. Proceedings of International Conference on Artificial Neural Networks (ICANN'95). pp. 53–60.

Leclerc, F., Plamondon, R., 1994. Automatic signature verification: the state of the art. International Journal on Pattern Recognition and Artificial Intelligence 8, 643–660.

Lee, S., Pan, J.C., 1992. Offline tracing and representation of signatures. IEEE Transactions. Systems, Man, and Cybernetics 22, 755–771.

Lippmann, R.P., 1987. An introduction to computing with neural nets. IEEE ASSP Magazine, 4–28.

Looney, C.G., 1997. Pattern Recognition using Neural Networks: Theory and Algorithms for Engineers and Scientists. Oxford University Press, Oxford.

Madhvanath, S., Govindaraju, V., 2001. The role of holistic paradigms in handwritten word recognition. IEEE Transactions. Pattern Analysis and Machine Intelligence 23, 149–164.

Pavlidis, I., Mavuduru, R., Papanikolopoulos, 1994. Off-line recognition of signatures using revolving active deformable models. Proceedings of IEEE International Conference on Systems, Man and Cybernetics. pp. 771–776.

Pavlidis, I., Papanikolopoulos, N.P., Mavuduru, R., 1998. Signature Identification through the use of deformable structures. Signal Processing 71, 187–201.

Perez, A., Sanchez, A., Velez, J.F., 2004. Simplified stroke-based approach for off-line signature recognition. Proceedings of Second COST 275 Workshop—Biometrics on the Internet. pp. 89–94.

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

Plamondon, R., Shirari, S.N., 2000. On-line and off-line handwriting recognition: a comprehensive survey. IEEE Transactions. Pattern Analysis and Machine Intelligence 22, 63–84.

Pontil, M., Verri, A., 1998. Support vector machines for 3-D object recognition. IEEE Transactions. Pattern Analysis and Machine Intelligence 20, 637–646.

Ramesh, V.E., Murty, M.N., 1999. Off-line signature verification using genetically optimized weighted features. Pattern Recognition 32, 217–233.

Riba, J.R., et al., 2000. Method for invariant signature classification. Proceedings of 15th International Conference on Pattern Recognition 2, 953–956.

Samanta, B., Al-Balushi, K.R., Al-Araimi, S.A., 2003. Artificial neural networks and support vector machines with genetic algorithm for bearing fault detection. Engineering Applications of Artificial Intelligence 16, 657–665.

Scholkopf, B., Sung, K., Burges, C., Girosi, F., Niyogi, P., Poggio, T., Vapnik, V., 1996. Comparing support vector machines with Gaussian kernels to radial basis function classifiers. AI Memo No. 1599, MIT.

Sethi, K.I., Han, K., 1995. Use of local structural association for retrieval and recognition signature images. Proceedings of the SPIE 2490, 125–136.

SVC, 2004. First International Signature Verification Competition. http://www.cs.ust.hk/svc2004/download.html.

Vapnik, V., 1995. The Nature of Statistical Learning Theory. Springer, New York.

Velez, J.F., Sanchez, A., Moreno, A.B., 2003. Robust off-line signature verification using compression networks and positional cuttings. Proceedings of IEEE International Conference on Neural Networks for Signal Processing (NNSP '03). pp. 627–636.