INFORMATICA, 2009, Vol. 20, No. 1, 115–138

Gabor-Based Kernel Partial-Least-Squares Discrimination Features for Face Recognition *

Vitomir ŠTRUC, Nikola PAVEŠIĆ
Faculty of Electrical Engineering, University of Ljubljana,
Tržaška 25, SI-1000 Ljubljana, Slovenia
e-mail: vitomir.struc@fe.uni-lj.si, nikola.pavesic@fe.uni-lj.si

Received: November 2007; accepted: August 2008

Abstract. The paper presents a novel method for the extraction of facial features based on the Gabor-wavelet representation of face images and the kernel partial-least-squares discrimination (KPLSD) algorithm. The proposed feature-extraction method, called the Gabor-based kernel partial-least-squares discrimination (GKPLSD), is performed in two consecutive steps. In the first step a set of forty Gabor wavelets is used to extract discriminative and robust facial features, while in the second step the kernel partial-least-squares discrimination technique is used to reduce the dimensionality of the Gabor feature vector and to further enhance its discriminatory power. For optimal performance, the KPLSD-based transformation is implemented using the recently proposed fractional-power-polynomial models. The experimental results based on the XM2VTS and ORL databases show that the GKPLSD approach outperforms feature-extraction methods such as principal component analysis (PCA), linear discriminant analysis (LDA), kernel principal component analysis (KPCA) or generalized discriminant analysis (GDA) as well as combinations of these methods with Gabor representations of the face images. Furthermore, as the KPLSD algorithm is derived from the kernel partial-least-squares regression (KPLSR) model it does not suffer from the small-sample-size problem, which is regularly encountered in the field of face recognition.

Keywords: Gabor features, Kernel partial-least-squares, face recognition, XM2VTS database, ORL database.

1. Introduction

Face recognition (verification or identification) is one of the most active research areas in the field of biometrics. Among the numerous feature-extraction methods that have been proposed for face recognition, appearance-based methods, such as principal component analysis (PCA) or linear discriminant analysis (LDA), have been the dominant feature-extraction techniques for years.

The PCA or Eigenface technique, introduced by Turk and Pentland (1991), applies the Karhunen–Loève transform to a set of training images and derives a number of projection

*Partially supported by the National Research Program P2-0250(C) Metrology and Biometric Systems, the Bilateral Project with the People's Republic of China Bi-CN/07-09-019, the Bilateral Project with the Bulgarian Academy of Sciences – Face and Signature Biometrics, the National Project AvID M2-0210 and the EU-FP7 Project 217762 Homeland Security, Biometric Identification and Personal Detection Ethics (HIDE).

axes (or eigenfaces), which act as the basis vectors for the PCA subspace. Each face image is represented as a vector of projection coefficients in this subspace, in which information compression, dimensionality reduction and de-correlation of the pixel values of the face images is achieved.

Unlike PCA, which strives to find a subspace that contains as much of the variance of the training samples as possible, LDA aims at constructing a linear subspace where the discriminative information between different classes of face images is emphasized. The objective is met through the maximization of the ratio of the between-class scatter matrix to the within-class scatter matrix, which serves as a criterion for class separability. It is generally believed that the incorporation of class-membership information into the feature-extraction technique improves the discrimination abilities of the extracted features. Thus, LDA is regularly found to out-perform PCA (see (Kittler *et al.*, 1999) for example).

However, due to the small number of available training images per subject the within-class scatter matrix of the training data is usually singular and LDA in its basic form is not applicable. This so-called *small-sample-size problem* (SSSP) demands a modification of the original LDA approach and is one of the most challenging problems in the field of face recognition. The traditional way of overcoming the SSSP in LDA is to project the training data into a lower-dimensional PCA subspace where the within-class scatter matrix is non-singular and then to perform the LDA in the reduced space. The described approach (often called the Fisherface approach (Belhumeur *et al.*, 1996)) provides an elegant solution to the SSSP and effectively uses the class-membership information to derive discriminative features for face recognition.

It is clear that PCA- and LDA-based approaches exhibit some fundamental differences that are crucial to the design of a face-recognition system. For example, PCA-based feature-extraction techniques do not suffer from the SSSP, but usually result in lower recognition rates than LDA-based techniques. LDA-based methods, on the other hand, ensure relatively high recognition rates, but even in their most elaborate form, require at least two training images per subject to be applicable. From the presented discussion we can see that an ideal feature-extraction technique should consider both the variability of the training data as well as the class-membership information and should furthermore be applicable in the most severe case of the SSSP, i.e., when only one training image per subject is available.

In addition to the SSSP, appearance-based methods should also be able to cope with nonlinear changes to face images that arise due to varying illumination, pose and facial expression during the image-acquisition phase. To that end, linear feature-extraction techniques such as PCA and LDA have been extended to their kernel (nonlinear) form by a number of researchers. Popular kernel methods that can be found in the literature include kernel principal component analysis (KPCA) (Schölkopf *et al.*, 1998), kernel fractional-step discriminant analysis (Dai *et al.*, 2007), generalized discriminant analysis (GDA) (Baudat and Anouar, 2000), kernel Fisher discriminant analysis (KFDA) (Yang, 2002) and kernel partial-least-squares discriminant analysis (KPDA) (Štruc *et al.*, 2008). All of these techniques have been shown to be superior to their linear counterparts in terms of face-recognition performance.

In recent years, another class of feature-extraction techniques attracted the interest of the face-recognition community – techniques based on the Gabor filter responses of face images. Unlike appearance-based methods, most Gabor-filter-based techniques use local characteristics of the face to derive discriminant features from the face image. This fact makes them robust in terms of illumination, pose and facial expression, but simultaneously susceptible to errors in the localization of the facial landmarks.

One of the first procedures to incorporate Gabor filters into the feature-extraction technique is the dynamic link architecture (DLA) introduced by Lades *et al.* in (Lades *et al.*, 1993). In DLA a rectangular graph is placed over the face image and a number of Gabor features (called Gabor jets) are extracted at each of the graph's nodes. Once a face graph is built, a graph-matching procedure is employed for recognition. Based on the dynamic link architecture, Wiskott *et al.* (1997) proposed a method called elastic bunch graph matching (EBGM), which improved upon DLA by introducing a face graph with nodes located at specific facial landmarks (fiducial points).

In contrast to the DLA and EBGM graph-based methods that consider only Gabor features extracted at a number of characteristic points, Liu and Wechsler (2002) proposed to use the whole face image for extraction of Gabor-based features. In their method a set of forty Gabor filters is employed to derive a high-dimensional feature vector and a variant of the multivariate linear discriminant analysis is used to reduce the vector's dimensionality. The presented approach, called the Gabor–Fisher Classifier (GFC), combined Gabor-filter-based methods with appearance-based techniques and proved to be highly effective in ensuring better recognition rates than the classical appearance-based or Gabor-based feature-extraction techniques on their own.

Researchers tried to improve upon the GFC technique by substituting the employed appearance-based method of the GFC with a more elaborate one. Liu (2004), for example, used kernel principal component analysis rather than LDA to derive the final facial features. Shen *et al.* (2007), on the other hand, applied the kernel form of LDA, i.e., generalized discriminant analysis, for the computation of the feature vectors. Although the listed methods exhibit some advantages over the GFC technique, they inherited the shortcomings intrinsic to the PCA- and LDA-based approaches.

To avoid the difficulties encountered when using (linear or nonlinear) PCA- and LDA-based feature-extraction techniques, we propose in this paper to employ a nonlinear variant of the partial-least-squares regression technique (called kernel partial-least-squares discrimination – KPLSD) for dimensionality reduction of the Gabor feature vector. As will be shown in the remainder of the paper, KPLSD has several advantages over the traditionally employed appearance-based methods. For example, like LDA-based methods, the KPLSD technique considers the class-membership information of the training data, while (like PCA-based approaches) it does not suffer from the SSSP, i.e., it is applicable even if only one training image per subject is available for training. Furthermore, experimental results on the XM2VTS and ORL databases show that the proposed approach, the Gabor-based kernel partial-least-squares discrimination (GKPLSD) method, outperforms similar methods described in the literature.

The rest of the paper is organized as follows. In Section 2, the theory behind Gabor wavelets and Gabor-based feature extraction is reviewed. The kernel partial-least-squares discrimination algorithm is presented in Section 3, and the novel Gabor-based KPLSD approach is detailed in Section 4. Experimental results based on the XM2VTS and ORL databases are given in Section 6 and 7. The paper concludes with some final comments in Section 8.

2. Gabor Feature Extraction

2.1. *Gabor Wavelets*

Since the discovery of the resemblance of Gabor wavelets (filters, kernels) and the 2D receptive field profiles of mammalian simple cells in the primary visual cortex, Gabor wavelet-based methods have been successfully employed in many computer-vision problems, ranging from fingerprint enhancement (Hong *et al.*, 1996) and texture segmentation (Bashar *et al.*, 2003) to face (Wiskott *et al.*, 1997) and palmprint recognition (Kong *et al.*, 2003). In addition to the similarity with the human visual system, Gabor wavelets also exhibit desirable characteristics of spatial locality and orientation selectivity, and are optimally localized in the space and frequency domains (Žibert and Mihelič, 2002). All of these properties make them a suitable choice for image decomposition and representation when the goal is to derive local and discriminating features (Liu, 2004).

In the spatial domain, a 2D Gabor wavelet is a Gaussian kernel function modulated by a complex plane wave (Daugman, 1985; Kyrki *et al.*, 2004):

$$\begin{aligned} \psi_{\Pi(f,\theta,\gamma,\eta)} &= \frac{f^2}{\pi\gamma\eta}\mathrm{e}^{-(\frac{f^2}{\gamma^2}x_t^2+\frac{f^2}{\eta^2}y_t^2)}\mathrm{e}^{j2\pi f x_t}, \\ x_t &= x\cos\theta + y\sin\theta, \\ y_t &= -x\sin\theta + y\cos\theta, \end{aligned} \tag{1}$$

where x and y denote the pixel coordinates, f denotes the frequency of the complex sinusoid, θ defines the orientation of the wavelet, γ is the spatial width of the wavelet along the sinusoidal plane wave and η is the wavelet's spatial width perpendicular to the wave.

To obtain an appropriate feature vector from an image, a bank of forty Gabor wavelets (five frequencies and eight orientations) is commonly created using the following parameters (Shen and Bai, 2006):

$$\begin{aligned} \psi_{g,h}(x,y) &= \psi_{\Pi(f_g,\theta_h,\gamma,\eta)}, \\ f_g &= f_{max}/\big(\sqrt{2}\big)^g, \\ \theta_h &= \frac{h}{8}\pi, \end{aligned} \tag{2}$$

where $\gamma = \eta = \sqrt{2}$, $f_{max} = 0.25$, $g \in \{0,\dots,4\}$ and $h \in \{0,\dots,7\}$.

Fig. 1 shows the real parts of Gabor wavelets for five scales $g \in \{0,\dots,4\}$ and eight orientations $h \in \{0,\dots,7\}$.

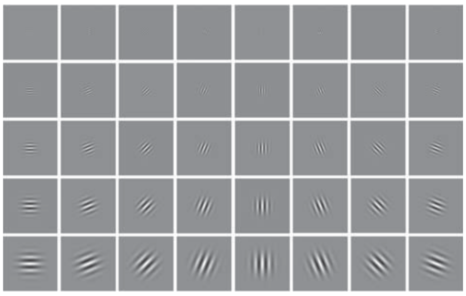

Fig. 1. Real parts of the Gabor wavelets for five scales and eight orientations.

2.2. *Gabor Feature Representation of Face Images*

The Gabor feature representation of a grey-scale face image $I(x, y) \in \mathbb{R}^{a\times b}$, where a and b are image dimensions (in pixels), is derived by convolving the image with the bank of Gabor wavelets, i.e.,

$$O_{g,h}(x, y) = I(x, y) * \psi_{g,h}(x, y), \tag{3}$$

where $O_{g,h}(x, y) \in \mathbb{R}^{a\times b}$ denotes the output of the convolution of a face image $I(x, y)$ with a Gabor wavelet $\psi_{g,h}(x, y)$ at scale g; ($g \in \{0, \dots, 4\}$) and orientation h; ($h \in \{0, \dots, 7\}$).

After convolving a face image $I(x, y)$ with all the Gabor wavelets $\psi_{g,h}(x, y)$ from the bank, the Gabor feature vector **x** is obtained by downsampling the magnitudes of $O_{g,h}(x, y)$ for all scales and orientations by a factor ρ, normalizing the results to have zero mean and unit variance, turning them into row vectors $\mathbf{o}^{\rho}_{g,h}$ and finally concatenating them into the feature vector **x** (Liu and Wechsler, 2002; Liu, 2004; Shen and Bai, 2006), i.e.,

$$\mathbf{x} = \left(\mathbf{o}^T_{0,0}\ \mathbf{o}^T_{0,1}\ \cdots\ \mathbf{o}^T_{4,7}\right)^T \in \mathbb{R}^N, \tag{4}$$

where $N = 40ab/\rho$.

It should be noted that the Gabor feature vector (4) consists only of the magnitudes of the convolution outputs, while the phase outputs have been discarded. Fig. 2 shows the magnitudes of the convolution outputs of a sample image from the XM2VTS database with the bank of Gabor wavelets whose real parts are presented in Fig. 1.

Despite the downsampling of the magnitudes of $O_{g,h}(x, y)$; ($g \in \{0, \dots, 4\}$, $h \in \{0, \dots, 7\}$), the resulting Gabor feature vector still resides in a very high-dimensional space. To further reduce its dimensionality and to enhance its discriminatory power, subspace projection techniques are commonly employed (Liu and Wechsler, 2002; Liu, 2004; Shen and Bai, 2006). One such technique, which has not previously been considered for the purposes of face recognition, is introduced in the following section.

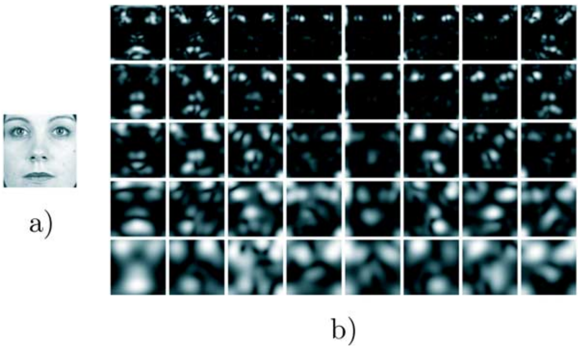

Fig. 2. Outputs of the convolution of a sample image from the XM2VTS database with the Gabor wavelets (Fig. 1) from the bank. (a) Sample image. (b) Magnitudes of the convolution outputs.

3. Partial-Least-Squares Regression

Partial-least-squares regression (PLSR), first introduced by Wold (1966) and later extended to its kernel form (KPLSR) by Rosipal and Trejo (2001), is a powerful statistical regression technique that can be used for relating several dependent variables (responses) to a large number of independent variables (predictors). PLSR models the relationship between the dependent and the independent variables by creating latent vectors (components, factors) in such a way that these vectors account for as much of the covariance between the dependent and independent variables as possible.

3.1. *Kernel Partial-Least-Squares Regression*

The PLSR model $\mathbf{Y} = \mathbf{XA} + \mathbf{B}$, where $\mathbf{A}$ is the matrix of the regression coefficients and $\mathbf{B}$ stands for the matrix of residuals, is constructed from a training set consisting of n observations (objects) with N x-variables (inputs), and M y-variables (responses).

Let $\mathbf{X}$ represent a $n \times N$ matrix of n N-dimensional inputs $\mathbf{x}$ and similarly let $\mathbf{Y}$ represent a $n \times M$ matrix of the corresponding n M-dimensional responses $\mathbf{y}$. PLSR is based on the decomposition of the matrices $\mathbf{X}$ and $\mathbf{Y}$ into the following form (Rosipal, 2003):

$$\begin{aligned} &\mathbf{X} = \mathbf{TP}^T + \mathbf{E}, \\ &\mathbf{Y} = \mathbf{UQ}^T + \mathbf{F}, \end{aligned} \tag{5}$$

where $\mathbf{T}$ and $\mathbf{U}$ are $n \times r$ matrices containing r latent vectors for n inputs, $\mathbf{P}$ and $\mathbf{Q}$ denote $N \times r$ and $M \times r$ matrices of weights (used to determine the linear combinations of inputs and responses, respectively) and the $n \times N$ and $n \times M$ matrices $\mathbf{E}$ and $\mathbf{F}$ represent matrices of residuals.

The parameters of the decomposition (5) are commonly determined by using the nonlinear iterative partial-least-squares (NIPALS) algorithm (Wold, 1966), which calculates the latent vectors $\mathbf{t}$ and $\mathbf{u}$ as well as the weight vectors $\mathbf{p}$ and $\mathbf{q}$ in such a way that the covariance between the $\mathbf{X}$ and $\mathbf{Y}$ matrices is maximized. (Note that the vectors $\mathbf{t}$, $\mathbf{u}$, $\mathbf{p}$ and $\mathbf{q}$ are columns of the matrices $\mathbf{T}$, $\mathbf{U}$, $\mathbf{P}$ and $\mathbf{Q}$, respectively.)

To extend the PLSR model (which accounts only for the second-order statistics in the data) to its nonlinear form (kernel partial-least-squares regression or KPLSR), Rosipal and Trejo (2001) proposed a modification of the NIPALS algorithm based on the "kernel trick", which avoids the calculation of the potentially computationally expensive nonlinear transformation Φ (where Φ denotes a nonlinear mapping of the N-dimensional input variable $\mathbf{x}$ from the input space $\mathbb{R}^N$ to a high-dimensional feature space $\mathcal{F}$, i.e., Φ: $\mathbf{x} \in \mathbb{R}^N \rightarrow \Phi(\mathbf{x}) \in \mathcal{F}$), and constructs the KPLSR model using the $n \times n$ kernel matrix

$$\mathbf{K} = \left[\Phi(\mathbf{x}_i)\Phi(\mathbf{x}_j)^T\right] = \left[K(\mathbf{x}_i, \mathbf{x}_j)\right]; \quad \forall i, j. \tag{6}$$

The NIPALS algorithm can be modified for use with such kernel matrices in two different ways: the first, in detail presented in (Rosipal and Trejo, 2001), is based on the same methodology as that used for deriving kernel PCA (Schölkopf *et al.*, 1998) and the second, introduced in (Bennett and Embrechts, 2003), uses direct factorization of the kernel matrix to produce its low-rank approximation. Considering the latter approach, the NIPALS algorithm is modified as follows (Bennett and Embrechts, 2003):

```
Input: Matrices $\mathbf{K}_c$ and $\mathbf{Y}$
Output: $r$ $n$-dimensional latent vectors $\mathbf{t}$ and $\mathbf{u}$
1. Randomly initialize the y-latent vector $\mathbf{u}$ (usually the first column of $\mathbf{Y}$).
2. Calculate the x-latent vector:
       $\mathbf{t} = \mathbf{K}_c\mathbf{K}_c^T\mathbf{u}, \quad \mathbf{t} \leftarrow \mathbf{t}/\|\mathbf{t}\|.$
3. Update the y-latent vector:
       $\mathbf{u} = \mathbf{K}_c\mathbf{K}_c^T\mathbf{t}, \quad \mathbf{u} \leftarrow \mathbf{u}/\|\mathbf{u}\|.$
4. Repeat Steps 2–3 until convergence (i.e., vector $\mathbf{u}$ in two consecutive iterations
   does not change).
5. Deflate the matrices $\mathbf{K}_c$ and $\mathbf{Y}$:
       $\mathbf{K}_c = \mathbf{K}_c - \mathbf{t}\mathbf{t}^T\mathbf{K}_c, \quad \mathbf{Y} = \mathbf{Y} - \mathbf{t}\mathbf{t}^T\mathbf{Y}.$
6. Orthonormalize the matrix Y: $\mathbf{Y} = \mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1/2}$.
7. Continue with Step 2. using the new matrices $\mathbf{K}_c$ and $\mathbf{Y}$.
```

Here $\mathbf{K}_c$ denotes the centered version of the kernel matrix $\mathbf{K}$, i.e., (Schölkopf *et al.*, 1998),

$$\mathbf{K}_c = \left(\mathbf{I} - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T\right)\mathbf{K}\left(\mathbf{I} - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T\right), \tag{7}$$

where $\mathbf{I}$ represents an n-dimensional identity matrix and $\mathbf{1}_n$ stands for a vector of all ones, with length n.

The algorithm is repeated until a sufficient number of latent vectors r is found. As stated in (Rosipal and Trejo, 2001), cross-validation should be employed to determine the optimal number of latent vectors r that ensures good generalization capabilities of the resulting KPLSR model.

After the matrices **T** and **U** are calculated, the predictions on N-dimensional inputs from the training set (matrix **X**) are made by

$$\widehat{\mathbf{Y}} = \mathbf{K}_c\mathbf{A}, \tag{8}$$

where $\mathbf{K}_c$ denotes the centered kernel matrix of the training data, $\widehat{\mathbf{Y}}$ denotes the matrix of the corresponding response estimates and

$$\mathbf{A} = \mathbf{K}_c^T\mathbf{U}(\mathbf{T}^T\mathbf{K}_c\mathbf{K}_c^T\mathbf{U})^{-1}\mathbf{T}^T\mathbf{Y} \tag{9}$$

represents the $n \times M$ matrix of the regression coefficients of the KPLS regression model.

Similarly, the predictions on the (new, test, query) inputs $\mathbf{X}^*$ are computed using

$$\widehat{\mathbf{Y}}^* = \mathbf{K}_c^*\mathbf{A}, \tag{10}$$

where

$$\mathbf{K}_c^* = \Big(\mathbf{K}^* - \frac{1}{n}\mathbf{1}_{n^*}\mathbf{1}_n^T\mathbf{K}\Big)\Big(\mathbf{I} - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T\Big) \tag{11}$$

is the centered version of the kernel matrix $\mathbf{K}^*$ of the inputs $\mathbf{X}^*$, n^* denotes the number of (new) inputs (i.e., the number of columns in $\mathbf{X}^*$) and $\mathbf{1}_{n^*}$ stands for a n^*-dimensional vector of all ones.

In this paper three kinds of kernels are considered for the construction of the KPLSR model, i.e., the Gaussian or rbf kernel (12), the polynomial kernel (13) and the recently proposed fractional-power-polynomial model ($fppm$) (14) (Liu, 2004), which has been shown to be particularly successful when deriving discriminative features from Gabor filtered images:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathrm{e}^{(\|\mathbf{x}_i - \mathbf{x}_j\|^2/2\sigma^2)}, \tag{12}$$

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T\mathbf{x}_j)^d, \tag{13}$$

$$K(\mathbf{x}_i, \mathbf{x}_j) = sign(\mathbf{x}_i^T\mathbf{x}_j)(|\mathbf{x}_i^T\mathbf{x}_j|)^z, \tag{14}$$

where $\sigma > 0$, $d \in \mathbb{N}^+$ and $0 < z < 1$.

3.2. *Kernel Partial-Least-Squares Discrimination*

PLSR and KPLSR can also be used for the classification of the independent variables if the matrix containing the dependent variables encodes the class membership of the independent variables. In this case, the matrix $\mathbf{Y}$ takes the following form (Barker and Rayens, 2003; Rosipal, 2003; Baek and Kim, 2004)

$$\mathbf{Y} = \begin{bmatrix} \mathbf{1}_{m_1} & \mathbf{0}_{m_1} & \cdots & \mathbf{0}_{m_1} \\ \mathbf{0}_{m_2} & \mathbf{1}_{m_2} & \cdots & \mathbf{0}_{m_2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{m_M} & \mathbf{0}_{m_M} & \cdots & \mathbf{1}_{m_M} \end{bmatrix}, \tag{15}$$

where M represents the number of classes in the set of n N-dimensional inputs (matrix $\mathbf{X}$), m_i represents the number of inputs in class C_i, $\mathbf{1}_{m_i}$ $(i = 1, 2, \ldots, M)$ denotes a $m_i \times 1$ vector of all ones and $\mathbf{0}_{m_i}$ $(i = 1, 2, \ldots, M)$ is a $m_i \times 1$ vector of all zeros.

Each of the rows in the matrix $\mathbf{Y}$ represents the desired regression response for the corresponding independent variable. The first m_1 rows, for example, define the desired regression responses for class C_1, the next m_2 rows define the regression responses for class C_2 and so forth. Thus, if the matrix in (15) is used for the computation of the KPLSR transformation matrix defined by (9), then the response estimates matrix $\widehat{\mathbf{Y}}$ in (8) serves as the foundation for constructing prototype models for all of the C_i classes $(i = 1, 2, \ldots, M)$. Here, each class is represented by the mean vector of the corresponding response estimates (features). Thus, the prototype model[2] for the ith class is computed as follows:

$$\bar{\mathbf{y}}_i = \frac{1}{m_i} \sum_{j=1}^{n} \widehat{\mathbf{y}}_j, \quad \forall \widehat{\mathbf{y}}_j \leftarrow C_i, \tag{16}$$

where $i = 1, 2, \ldots, M$; $j = 1, 2, \ldots, n$; $\widehat{\mathbf{y}}_j$ stands for the jth row of the response estimates (i.e., feature) matrix $\widehat{\mathbf{Y}}$ and the expression $\widehat{\mathbf{y}}_j \leftarrow C_i$ indicates that only the response-estimate vectors $\widehat{\mathbf{y}}_j$ that correspond to class C_i should be used for the calculation of the ith prototype model $\bar{\mathbf{y}}_i$.

When a new (test) input $\mathbf{X}^*$ needs to be classified, its regression-response estimate (i.e., its feature vector) $\widehat{\mathbf{Y}}^*$ is first computed and then matched to the appropriate class prototype. Depending on the outcome of the matching procedure the input is ultimately classified into the adequate class (a more detailed description of the matching procedure will be given in Section 5).

As the goal of the presented technique is classification rather than regression, it is usually referred to as kernel partial-least-squares discrimination (KPLSD).

4. The Gabor-Based Kernel Partial-Least-Squares Discrimination Feature-Extraction Algorithm

In this section we present the novel facial-feature-extraction method that combines the Gabor feature-extraction approach described in Section 2 with the kernel partial-least-

[2] In the field of biometrics usually called template or client model

squares discrimination (KPLSD) algorithm presented in Section 3. Our proposal is based on the following three facts:

- the Gabor wavelet representation of a face image contains discriminative information that is considered to be robust against changes in illumination, pose and facial expression (Liu, 2004; Shen and Bai, 2006),
- KPLSD does not suffer from the small-sample-size problem (i.e., it is specifically designed to deal with situations where the number of training samples n is smaller than the dimensionality of the samples N),
- like discriminative methods, such as LDA or GDA, KPLSD uses the class-membership information of the training data for the derivation of the facial features and is therefore expected to outperform expressive feature-extraction techniques such as PCA or KPCA.

4.1. *The GKPLSD Algorithm*

The GKPLSD algorithm can be summarized as follows:

Input: A database of face images containing images from three sets: training $\mathcal{X}$, evaluation $\mathcal{E}$ and test $\mathcal{T}$.

Output: A set of low-dimensional face-feature vectors $\mathcal{Y} = \{\widehat{\mathbf{y}}_i \in \mathbb{R}^M\colon\ i = 1, 2, \ldots\}$.

1. Convolve a face image from the given database with the set of forty Gabor wavelets (3).
2. Downsample the magnitudes of the convolution outputs by the factor ρ.
3. Normalize the downsampled magnitudes of the convolution outputs to have zero mean and unit variance.
4. Combine the normalized and downsampled magnitudes of the convolution outputs (4).
5. Repeat Steps 1–4 for each face image in the database.
6. Compose matrices $\mathbf{K}$ (6), $\mathbf{K}_c$ (7) and $\mathbf{Y}$ (15) using the Gabor feature vectors from the training set.
7. Compute r latent vectors **t** and **u** using the modified NIPALS algorithm.
8. Compute the transformation matrix **A** (9).
9. Compute the matrix of low-dimensional face-feature vectors $\widehat{\mathbf{Y}}$ (8).
10. Compute the kernel matrices $\mathbf{K}^*$ (6) and $\mathbf{K}_c^*$ (11) using the Gabor feature vectors of the face images from the evaluation and test sets of the database.
11. Compute the matrix of low-dimensional face-feature vectors $\widehat{\mathbf{Y}}^*$ (10).

4.2. *Computational Complexity of the GKPLSD Algorithm*

To analyze the computational complexity of the proposed GKPLSD algorithm, we will treat the Gabor filtering and the KPLSD algorithm as two separate techniques. This will ensure that the presented computational cost will be easy to compare with that of other techniques that combine Gabor-based features with appearance-based methods (e.g., the GFC method). Moreover, as most of the computational complexity of the GKPLSD technique falls into the training stage, we will focus primarily on this part of the method. The cost for applying GKPLSD to a test sample, can, however, be deduced from the discussion in the remainder of this section.

It is known that image filtering can be performed in two different ways: either with the help of expression (3), i.e., by convolving the image with the given filter in the spatial domain, or by a simple pixel-wise multiplication in the Fourier domain. For a $P \times P$ image and a $F \times F$ filter the first way results in a computational complexity of $\mathrm{O}(F^2)$ for each image pixel. Hence, for the whole image $\mathrm{O}(P^2F^2)$ multiplications and/or additions (MADDs) are needed. If we assume that the available training set comprises n such images, the total computational cost for performing Gabor filtering with all 40 filters in the spatial domain is $\mathrm{O}(40nP^2F^2)$. For the second option, i.e., filtering in the frequency domain, the computational complexity is independent of the size of the applied filter, but depends heavily on the dimensions of the image to be filtered. In this case the cost of performing a filtering operation on the image is $\mathrm{O}(P^2 \log_2 P)$, thus, for the total training set and the whole filter bank it equals $\mathrm{O}(40nP^2 \log_2 P)$ (Fialka and Cadik, 2006). It should be noted that for large filter kernels, such as the Gabor filters, the second option is much faster and should be preferred to the first one.

As already mentioned above, most of the computational complexity of the proposed GKPLSD technique falls into the training stage, mainly due to the need for computing the KPLSD transformation matrix (9), which requires calculating the kernel matrix and employing the NIPALS algorithm. Computing the kernel matrix has a complexity of at least $\mathrm{O}(Nn^2)$, where n again denotes the number of training samples (e.g., images) in the training set and N represents the number of elements (e.g., pixels) in each of the samples (Yang *et al.*, 2004). Applying the NIPALS algorithm, on the other hand, requires $\mathrm{O}(rn^3)$ operations, where r represents the number of desired latent vectors. As stated in (Nicolai *et al.*, 2007), each iteration of the NIPALS algorithm has a complexity of $\mathrm{O}(n^3)$, while the total cost is also directly proportional to the number of desired latent vectors r. Once the KPLSD model is built (i.e., the NIPALS algorithm has been applied to the training data) each of the training samples (or test samples) is projected into the KPLSD subspace – a procedure that requires $\mathrm{O}(nN + nM)$ operations, where M denotes the number of classes in the training data (i.e., $\mathrm{O}(nN)$ operations are required to construct the test kernel matrix for the given sample and $\mathrm{O}(nM)$ operations for the computation of the final features – the KPLSD projection).

As all kernel methods require the computation of at least one kernel matrix and some kind of eigen-analysis, the overall computational cost of KPLSD is comparable to that of other kernel methods such as KPCA or GDA.

4.3. *Discussion of the GKPLSD Algorithm*

An important factor when choosing a feature-extraction approach for a face-recognition system is the computational characteristics of the employed technique. In this section we will compare the characteristics of the proposed method with those of some similar approaches found in the literature. As Gabor filtering is a well-studied topic (see, for example, (Liu and Wechsler, 2002; Liu, 2004; Kyrki *et al.*, 2004; Shen and Bai, 2006; Shen *et al.*, 2007)), we will confine ourselves to a comparison of the second part of the proposed Gabor-based kernel partial-least-squares discrimination method, i.e., the KPLSD algorithm.

So far we have pointed out several times that, like the PCA-based feature-extraction techniques, the proposed KPLSD method does not suffer from the SSSP, while it still (like the discriminative approaches) considers the class-membership information of the training data and should therefore result in better recognition performance than PCA-based methods. However, a couple of questions arise from the above statement: first, why should the class-membership information be incorporated into the feature-extraction technique, and second, do the PLSD and KPLSD methods improve upon the traditionally used discriminative methods such as LDA or GDA?

To answer the first question, we have to analyze the expressive methods (e.g., PCA) in more detail. The goal of PCA is to find a subspace whose basis vectors correspond to the maximum variance directions present in the training data. Each training sample can be projected into this subspace and again reconstructed with minimum error. As stated in (Draper *et al.*, 2003), PCA is an optimal compression scheme that minimizes the mean squared error between the original training data and its reconstruction for any given level of compression. From this point of view, it is obvious that PCA is tailored towards data representation and reconstruction rather than classification (similar observations could also be made for the kernel form of PCA – KPCA). Thus, for a feature-extraction technique to be suitable for the recognition task, classification-specific information, such as the class-membership of the training data, should be incorporated into the technique.

To answer the second question, we have to examine the shortcomings of the traditionally used discriminative methods, LDA and GDA, and analyze our findings with respect to the KPLSD approach. The LDA and GDA techniques seek projection directions, either in the original data space or in the high-dimensional Hilbert space, which discriminate well between training samples of the different classes. However, as they accomplish this task through an eigen-analysis of the scatter matrices (or the kernel forms of the scatter matrices), they are susceptible to the SSSP, which results in the (kernel) within-class scatter matrix being singular. Hence, the LDA and GDA methods, which require an inverse of the (kernel) within-class scatter matrix to be applicable, need modifications to handle the SSSP and consequently result in sub-optimal projection directions with respect to the original discrimination criterion. KPLSD, on the other hand, requires no inversion of any kernel matrix; instead it considers the class-membership information implicitly through the predesigned response matrix $\mathbf{Y}$ in (15). Furthermore, as pointed out in (Loog *et al.*, 2001) and (Dai *et al.*, 2007), LDA and GDA tend to overemphasize inter-class distances between classes that are already well separated in the original data space. Of course, this is at the expense of the classes that are close to each other. This usually leads to a large overlap of the last-mentioned classes in the final LDA and GDA feature spaces. Again, KPLSD avoids the presented difficulty with the help of the predesigned response matrix $\mathbf{Y}$, which ensures that the projections (i.e., the feature vectors) of the training data from different classes are distanced equally, one from the other, in the KPLSD subspace.

5. Matching, Decision and Errors

The proposed feature-extraction technique (GKPLSD) was tested within a face-recognition system in the verification and identification modes.

In the verification mode the claim of the identity (the entry of the person's name or the person's identification number) of a person presented to the system results in the acquired face image being transformed into the "live" feature vector, and matched with the claimed person's prototype model (in our case the mean feature vector of the face images acquired during enrolment). If the person's "live" feature vector and the mean feature vector of the claimed identity have a degree of similarity that is higher than the system threshold, then the claim is accepted; otherwise, the claim is rejected (or vice versa if a dissimilarity measure is used). Hence, a face-recognition system operating in verification mode aims at verifying the validity of the identity claim (Chen *et al.*, 2007).

In the identification mode the person currently presented to the system does not have to explicitly claim an identity; instead, the system performs a comparison of the "live" feature vector and the prototype models of all the enrolled users to find a match. In practice, the system establishes a person's identity by searching for the prototype model that best matches the "live" feature vector (i.e., in the case that dissimilarity measures are used for the matching-score calculation, the system searches for the prototype model that results in the lowest matching score among all the examined prototypes) and consequently assigns the identity corresponding to that prototype model to the "live" feature vector.

Obviously, a face-recognition system (operating in verification or identification mode) requires a scoring function to be able to perform feature matching. To that end, three commonly used dissimilarity measures are considered for the calculation of the matching scores in this paper, namely:

- the L_1 distance

$$d_{L_1}(\widehat{\mathbf{y}}, \bar{\mathbf{y}}_i) = |\widehat{\mathbf{y}} - \bar{\mathbf{y}}_i|, \tag{17}$$

- the L_2 distance

$$d_{L_2}(\widehat{\mathbf{y}}, \bar{\mathbf{y}}_i) = (\widehat{\mathbf{y}} - \bar{\mathbf{y}}_i)^T(\widehat{\mathbf{y}} - \bar{\mathbf{y}}_i), \tag{18}$$

- the cosine distance

$$d_{cos}(\widehat{\mathbf{y}}, \bar{\mathbf{y}}_i) = 1 - \widehat{\mathbf{y}}^T\bar{\mathbf{y}}_i/\|\widehat{\mathbf{y}}\|\,\|\bar{\mathbf{y}}_i\|, \tag{19}$$

where $\widehat{\mathbf{y}}$ denotes a face-feature vector, $\bar{\mathbf{y}}_i$ denotes the mean feature vector of the system client C_i, $(i = 1, 2, \ldots, M)$, and M denotes the number of system clients.

To measure the effectiveness of the proposed feature-extraction technique and to assess its impact on the system's verification and identification performance, several standard error and recognition rates (presented in the remainder of the section) are used in this paper.

For each series of verification experiments three error rates are computed. The first, called the false acceptance rate (FAR), indicates the number of incorrectly accepted impostors, while the second, called the false rejection rate (FRR), indicates the number of incorrectly rejected clients (Su, 2007). The last error rate – the total error rate (TER = FAR + FRR) – is used as a comparison metric for different verification experiments (Mihelič and Žibert, 2006).

For the identification experiments the results are presented in terms of the rank-one recognition rate, which measures the percentage of tested face images that were correctly identified, i.e., the prototype models corresponding to the assigned identity resulted in the best match with the "live" feature vector.

6. Verification Experiments

6.1. *The XM2VTS Database and Experimental Setup*

The verification experiments were performed on frontal face images from the multimodal XM2VTS database (Messer *et al.*, 1999), which contains audio and video data as well as color images of 295 individuals, captured in four separate sessions. As two recordings were made during each session, a total of 2360 frontal face images (8 per subject) were available for the training and the testing of the GKPLSD feature-extraction algorithm.

All the face images from the database were appropriately pre-processed and normalized. The pre-processing included converting the input color images into 8-bit grey-scale images, manually locating the eyes, geometrically normalizing (rotating and scaling) the images in such a way that the centers of the eyes were located at predefined positions, cropping the face parts of the images to a standard size of 128 × 128 pixels, and finally photometrically normalizing the face images by removing their mean and scaling their pixels to unit variance (Ribarić *et al.*, 2008). Some examples of the normalized images from the database are shown in Fig. 3.

The first configuration of the Lausanne protocol (LP1) (Luettin and Maitre, 1998) was used to divide the 295 subjects of the XM2VTS database into groups of clients (200 subjects) and impostors (95 subjects), and to distribute their images among three sets that were employed for training (3 images per subject), evaluating and testing the feature-extraction method GKPLSD in face-verification experiments.

The pre-processed and normalized face images from the XM2VTS database were subjected to the GKPLSD feature-extraction method, as described in Section 4. In all experiments the down-sampling factor was set to 32 ($\rho = 32$), and the number of features in the low-dimensional feature vectors was set to its maximal value, i.e., to the number

Fig. 3. Examples of pre-processed images from the XM2VTS database.

of client classes ($M = 200$). Note that in contrast to the feature-extraction methods presented in (Liu and Wechsler, 2002; Liu, 2004), where the down-sampling factor was set to 64, a two-times smaller down-sampling factor was used in this paper to retain more discriminative information in the Gabor feature vectors.

6.2. *Tuning the KPLSD Model's Parameters*

The goal of the first set of verification experiments was to determine the number of latent vectors r (see Section 3 for details) that would ensure optimal generalization abilities of the KPLSD model and to select the most appropriate kernel function to be used in conjunction with the proposed GKPLSD feature-extraction method. Five types of kernel functions were considered for comparison, i.e., the Gaussian or *rbf* kernel (12) for $\sigma = 200$, polynomial kernels (13) for d =1, 2 and 3, and the $fppm$ kernel (14) for $z = 0.9$ (note that the parameters σ and z were chosen empirically). The verification performance of the GKPLSD with each of the listed kernels was tested for four different numbers of latent vectors r (i.e., $r = 50$, $r = 150$, $r = 250$ and $r = 350$). The results (error rates FAR, FRR and TER) presented in Table 1 were obtained using the the evaluation images set and the L_2 distance measure as the scoring function.

It is clear from Table 1 that the lowest verification error (TER) for all the tested kernel functions was obtained when 250 latent vectors were used for construction of the the KPLSD model. The reason for such behavior is that models considering only 50 or 150 latent vectors could not capture enough of the information contained in the Gabor feature vectors (4), whereas 350 latent vectors resulted in an overfitted KPLSD model. Among the tested kernel functions, the $fppm$ ($z = 0.9$) performed slightly better than the rbf ($\sigma = 200$) and the first-degree polynomial kernel. Using second- or third-degree polynomial kernels led to a reduced performance of the proposed feature-extraction method when compared to the $fppm$, rbf or first-degree polynomial kernels.

The presented findings suggest that fractional-power-polynomial models and 250 latent vectors should be used with the GKPLSD method to achieve optimal verification performance.

6.3. *Comparison of Different Distance Measures*

The second series of verification experiments assessed the performance of the GKPLSD feature-extraction method with different distance measures at the matching stage. As

Table 1

Verification performance of the GKPLSD for different numbers of latent vectors and different kernel functions (evaluation set)

No. of loadings	Error rate	Kernel function				
		poly 1	*poly 2*	*poly 3*	*rbf*	***fppm***
	FAR(%)	4.62	4.90	10.37	4.20	4.99
$r = 50$	*FRR*(%)	4.50	4.83	9.83	4.00	4.67
	TER(%)	9.12	9.73	20.20	8.20	9.66
	FAR(%)	5.22	8.18	12.56	5.58	5.22
$r = 150$	*FRR*(%)	5.17	8.17	12.17	5.50	5.17
	TER(%)	10.39	16.35	24.73	10.08	10.39
	FAR(%)	3.21	3.82	4.36	3.07	2.78
***r* = 250**	*FRR*(%)	3.00	3.50	4.33	2.83	2.67
	TER(%)	6.21	7.32	8.69	5.90	**5.45**
	FAR(%)	4.26	5.68	6.36	4.50	4.39
$r = 350$	*FRR*(%)	4.17	5.67	6.17	4.59	4.33
	TER(%)	8.43	11.35	12.53	9.09	8.72

mentioned in Section 5, the distance measure d_{L_1} (17), the distance measure d_{L_2} (18) and the cosine distance measure d_{cos} (19) were used in the comparison. The KPLSD algorithm was implemented with a fractional-power-polynomial model (for $z = 0.9$) and 250 latent vectors.

A comparison of the generated ROC curves for the tested distance measures is shown in Fig. 4. From the graphs it is clear that the cosine distance measure performed the best, followed in order by the distance measures d_{L_2} and d_{L_1}. Similar results were also obtained on face images from the test set (see Table 2), where d_{cos} outperformed the other two measures by a large margin.

Based on the results we can conclude that the optimal distance (or dissimilarity) mea-

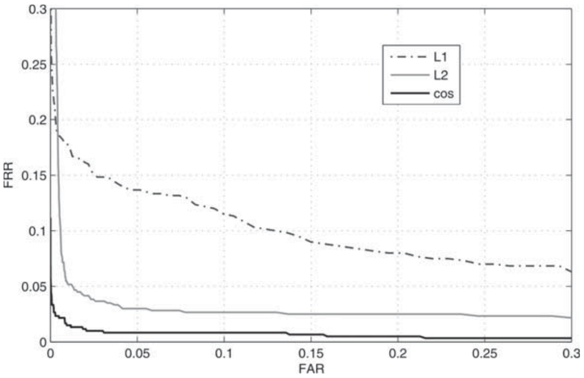

Fig. 4. ROC curves of the GKPLSD feature-extraction method for different distance measures (evaluation set).

Table 2

Verification performance of the GKPLSD feature-extraction method with different distance measures (test set)

Distance measure	*FAR*(%)	*FRR*(%)	*TER*(%)
d_{L_1}	8.21	13.75	21.96
d_{L_2}	3.08	2.25	5.33
d_{cos}	0.76	1.50	**2.26**

sure to be used in conjunction with the proposed GKPLSD feature-extraction method is the cosine distance measure.

6.4. *Comparison with Other Face-Feature-Extraction Methods*

In the last set of our face-verification experiments we compared the performance of the proposed GKPLSD method with that of other, popular techniques widely used for face verification. Specifically, the following methods were implemented for comparison: PCA (Turk and Pentland, 1991), LDA or Fisherfaces (Belhumeur *et al.*, 1996), KPCA (Schölkopf *et al.*, 1998), GDA (Baudat and Anouar, 2000), KPLSD (presented in this paper), Gabor-based PCA, Gabor-based KPCA (Liu, 2004), Gabor-based GDA (Shen *et al.*, 2007) and the Gabor-based LDA method presented in (Liu and Wechsler, 2002)[3]. The parameters (i.e., the number of features, the kernel function, the threshold, etc.) of the listed methods were all optimized to yield the lowest possible total error rate.

The results of the verification experiments are shown in Table 3. It is clear that the proposed Gabor-based KPLSD method performed best, followed in order by the Gabor-based GDA, Gabor-based LDA, the KPLSD, the GDA, the Fisherface, the Gabor-based KPCA, the Gabor-based PCA, the KPCA and the PCA methods. Generally, all the subspace techniques (linear and nonlinear) performed better when they were implemented on Gabor-filtered images instead of the original ones. An improvement in the total error rate of at least 30% was observed for all the methods.

Another interesting observation regarding the performance of the linear (LDA) and nonlinear discriminative (KPLSD and GDA) methods can be made with regard to the error rates in Table 3. While the total verification error of these methods is almost the same if the features are extracted from the original input images, the difference increases if Gabor-filtered images are used for the derivation of the final feature vectors. Here, the KPLSD feature-extraction technique seems to be particularly successful, as the total error rate is reduced by almost 50% when compared to the LDA approach. The reason for such behavior can be found in the fact that KPLSD is specifically designed to handle

[3] Note that unlike the KPLSD, KPCA and GDA (which are nonlinear kernel methods) techniques, LDA and PCA represent linear methods and usually serve as baseline techniques when comparing the verification performance of different feature-extraction approaches.

Table 3

Verification performance comparison for the different methods (test set)

Method	*FAR*(%)	*FRR*(%)	*TER*(%)
PCA	9.30	6.75	16.05
LDA	3.86	3.00	6.86
$KPCA$	7.65	6.25	11.13
GDA	3.52	2.75	6.27
$KPLSD$	3.37	2.75	6.12
$GaborPCA$	5.04	4.25	9.29
$GaborLDA$	1.91	2.50	4.41
$GaborKPCA$	3.84	5.00	8.84
$GaborGDA$	1.92	1.75	3.67
$GaborKPLSD$	0.76	1.50	**2.26**

over-determined problems (or small sample-size problems, where $n \ll N$)[4]. In contrast, LDA struggles with its generalization ability when only a small number of samples are available for training (Martinez and Kak, 2001).

7. Identification Experiments

7.1. *The ORL database and experimental setup.*

For our second class of face-recognition experiments, i.e., the identification experiments, we used the ORL database acquired at the Olivetti Research Laboratory in Cambridge, U.K. The database is made up of 400 face images that correspond to 40 distinct subject. Thus, each subject in the database is represented with 10 facial images that exhibit variations in terms of illumination, pose and facial expression. The images are stored at a resolution of 92×112 and 8-bit grey levels (Samaria and Harter, 1994).

Like the images from the XM2VTS database, the ORL images were subjected to our pre-processing procedure that, based on the manually marked eye locations, normalized (geometrically and photometrically) and cropped the face regions of all the face images to a standard size of 64×64 pixels.

For the identification experiments presented in Sections 7.2 to 7.3 the database was divided into two separate parts. The first part comprised three face images of each of the 40 subjects and was used for training, while the second part comprised the remaining seven images of the database's subjects and was used for testing. This setup resulted in a total of 120 face images available for training and 280 images that had to be identified. For the last series of identification experiments, however, a different setup was used and will be described in detail in Section 7.4.

[4] Note that the dimension of a Gabor feature vector for a 128×128 input image and a down-sampling factor of $\rho = 32$ is equal to $N = 20\,480$, while the dimensionality of the input vector derived from the original images is “only” $128 \times 128 = 16\,384$.

Table 4

Rank-one recognition rates (in %) of the GKPLSD for different numbers of latent vectors and different kernel functions

No. of loadings	Kernel function				
	poly 1	*poly 2*	*poly 3*	*rbf*	***fppm***
$r = 30$	83.6	68.2	46.8	83.6	84.3
$r = 50$	91.8	82.5	87.9	91.9	92.5
$\mathbf{r = 70}$	97.5	93.2	90.0	97.8	**97.9**
$r = 90$	95.4	92.1	88.9	95.3	96.1

7.2. *KPLSD Model Parameter Tuning*

Although face identification and verification are related problems we cannot simply generalize our findings presented in Section 6 and use the same model parameters for the identification case. Our first series of experiments aimed, therefore, at optimizing the parameters of the KPLSD model for the identification problem. Like with the verification experiments in Section 6 five kernel functions, i.e., the first-, second- and third-degree polynomial kernel functions, the rbf kernel (this time for $\sigma = 1000$) and the $fppm$ kernel for $z = 0.9$, were tested for their identification performance in conjunction with four different numbers of latent vectors, i.e., $r = 30$, $r = 50$, $r = 70$ and $r = 90$. The results of the experiments with the d_{L_2} distance as a scoring function are presented in Table 4.

From the results we can see that the best identification performance, i.e., 97.8%, was achieved with the KPLSD model constructed with a fractional power polynomial model and 70 latent components. This model should, therefore, be used for the following assessments of the GKPLSD approach on the ORL database.

7.3. *Comparison of Different Scoring Functions*

To assess the impact of different distance measures on the performance of the GKPLSD approach in the identification scenario our second series of identification experiments employed the three scoring functions presented in Section 5 for the matching-score calculation. From the results presented in Table 5 it is clear that like with the verification case, the cosine measure performed the best, followed in order by the d_{L_2} (18) and d_{L_1} (17) distance measures.

Table 5

Rank-one recognition rates (in %) of the GKPLSD feature-extraction method with different dissimilarity measures

Dissimilarity measure	d_{L_1}	d_{L_2}	d_{cos}
Rank one recognition rate (%)	97.4	97.9	98.2

Interestingly, while still being superior to the other two measures, the cosine measure only slightly improves the recognition rate when compared to the verification case. Nevertheless, the results suggest that the cosine dissimilarity measure should be used in conjunction with the GKPLSD approach in the final comparative assessment.

7.4. *Comparison with Popular Feature-Extraction Techniques*

In our last series of identification experiments the performance of the proposed KPLSD technique was compared to that of some popular feature-extraction techniques found in the literature. Again, the first two techniques were linear methods, i.e., PCA and LDA, which served as baseline approaches in the comparison. However, as the proposed KPLSD technique is a nonlinear (kernel) approach, two additional kernel methods, i.e., KPCA and GDA, were also implemented to allow for a fair comparison with our approach. All the listed methods were applied to the original as well as to the Gabor-filtered images.

For this set of experiments a different experimental setup was used than in the previous two sections. Here, the number of training images per subject was increased from one to nine, while the remaining images were used for testing. For example, if two training images per subject were used in the training stage, eight images were left available for determining the rank-one recognition rate of the given feature-extraction technique. Clearly, this setup resulted in more identification experiments being performed when fewer training images were used and, vice versa, fewer identification experiments were performed when more images were used for the training. Furthermore, as the recognition rate depends on the choice of the training and test sets, the experiments were repeated five times for a given number of training images (each time the training set was chosen randomly). The recognition rates presented in Table 6 represent, therefore, the average recognition rates over five repetitions of the identification experiments.

Table 6

Comparison of the average rank-one recognition rates (in %) of different feature-extraction techniques

Method	Number of training images								
	1	2	3	4	5	6	7	8	9
PCA	50.7	66.2	72.0	76.1	78.8	80.1	81.7	82.0	82.1
LDA	n/a	69.8	90.2	93.4	94.7	97.1	97.2	98.4	99.5
$KPCA$	51.9	67.6	75.6	79.3	82.4	81.6	83.0	84.0	86.0
GDA	n/a	82.0	91.1	94.4	95.2	97.8	97.3	98.5	99.0
$KPLSD$	65.2	82.1	91.5	95.4	95.0	98.2	98.0	98.8	100
$GaborPCA$	58.1	73.2	82.4	86.6	91.2	92.8	93.5	95.3	95.5
$GaborLDA$	n/a	77.4	95.1	98.0	98.9	99.5	99.7	100	100
$GaborKPCA$	68.4	84.5	89.7	92.2	95.3	98.1	97.7	97.5	99.5
$GaborGDA$	n/a	92.0	95.7	98.5	99.4	99.8	100	100	100
$GaborKPLSD$	75.9	92.4	97.7	99.2	99.3	99.8	100	100	100

The results show the true strength of the proposed GKPLSD approach. We can see that when only one face image per subject is available for training neither LDA nor GDA is applicable (denoted as n/a in Table 6). The KPLSD algorithm, on the other hand, is not only applicable, but also ensures significantly higher recognition rates than PCA or KPCA on the original as well as the Gabor-filtered images. If we consider the overall results, i.e., across different numbers of training images, we can see that again the proposed GKPLSD performed the best in almost all cases, while the relative ranking of the compared techniques is similar as in the comparative assessment for the face-recognition system operating in verification mode.

8. Conclusion

In this paper we have presented a novel Gabor-based KPLSD algorithm for extracting discriminative features from frontal face images. The effectiveness of the proposed algorithm was demonstrated in a series of face-recognition (verification and identification) experiments, performed on the XM2VTS and ORL databases. The best results were obtained when fractional-power-polynomial models were used for the implementation of the KPLSD part of the proposed feature-extraction method, and the cosine distance measure was employed at the matching stage for the calculation of the matching scores. In a comparative assessment, where in addition to the proposed GKPLSD method, widely used feature-extraction techniques such as PCA, LDA, KPCA, GDA and combinations of these techniques with Gabor-based features were tested for their verification and identification performances, the GKPLSD achieved the best results, while effectively avoiding the small sample size problem regularly encountered in the field of face recognition. Our future work with respect to the GKPLSD technique, more precisely the KPLSD algorithm, will be focused on examining the mathematical relations between KPLSD and other nonlinear discriminative techniques such as GDA.

References

Baek, J., and M. Kim (2004). Face recognition using partial least squares components. *Pattern Recognition*, **37**(6), 1303–1306.

Barker, M., and W.S. Rayens (2003). Partial least squares for discrimination. *Journal of chemometrics*, **17**, 166–173.

Bashar, M.K., T. Matsumoto and N. Ohnishi (2003). Wavelet transform-based locally orderless images for texture segmentation. *Pattern Recognition Letters*, **24**(15), 2633–2650.

Baudat, G., and F.E. Anouar (2000). Generalized discriminant analysis using a kernel approach. *Neural Copmutation*, **12**(10), 2385–2404.

Belhumeur, P.N., J.P. Hespanha and D.J. Kriegman (1996). Eigenfaces vs. fisherfaces: recognition using class specific linear projection. In *Proceedings of the 4th ECCV*, Cambridge. pp. 45–58.

Bennett, K.P., and M.J. Embrechts (2003). An optimization perspective on kernel partial least squares regression. In J.A.K. Suykens, G. Horvath, S. Basu, C. Micchelli and J. Vandewalle (Eds.), *Advances in Learning Theory*: *Methods, Models and Applications*. IOS Press Amsterdam, The Netherlands. pp. 227–249.

Chen, T.H., G. Horng and K.C. Wu (2007). A secure YS-like user authentication scheme. *Informatica*, **18**(1), 27–36.

Dai, G., D.Y. Yeung and Y.T. Qian (2007). Face recognition using kernel fractional-step discriminant analysis algorithm. *Pattern Recognition*, **40**(1), 229–243.

Daugman, J.G. (1985). Uncertainty Relation for resolution in space, spatial frequency, and orientation optimized by 2D cortical filters. *Journal of the Optical Society of America*, **2**(7), 1160–1169.

Draper, B.A., K. Baek, M.S. Bartlett and J.R. Beveridge (2003). Recognizing faces with PCA and ICA. *Computer Vision and Image Understanding*, **91**(1–2), 115–137.

Fialka, O. and M. Cadik (2006). FFT and image filtering performance on GPU. In *Proceedings of the 10th CIV*, London. pp. 609–614.

Hong, L., A. Jain, S. Pankanti and R. Bolle (1996). Fingerprint enhancement. In *Proceedings of the 1st IEEE WACV*, Sarasota. pp. 202–207.

Kong, W.K., D. Zhang and W. Li (2003). Palmprint feature extraction using 2-D Gabor filters. *Patern Recognition*, **36**(10), 2339–2347.

Kyrki, V., J.K. Kamarainen and H. Kalviainen (2004). Simple Gabor feature space for invariant object recognition. *Pattern Recognition Letters*, **25**(3), 311–318.

Lades, M., J.C. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, R.P. Wurtz and W. Konen (1993). Distortion invariant object recognition in the dynamic link architecture. *IEEE Transactions on Computers*, **42**(3), 300–311.

Li, Y., J. Kittler and J. Matas (1999). Effective implementation of linear discriminant analysis for face recognition and verification. In *Proceedings of the 8th CAIP*, Ljubljana. pp. 234–242.

Liu, C., and H. Wechsler (2002). Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(4), 467–476.

Liu, C. (2004). Gabor-based kernel PCA with fractional power polynomial models for face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **26**(5), 572–581.

Loog, M., R.P.W. Duin and R. Haeb-Umbach (2001). Multi-class linear dimension reduction by weighted pairwise Fisher criteria. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **23**(7), 762–766.

Luettin, J., and G. Maitre (1998). Evaluation protocol for the extended M2VTS database. *Tech. Report IDIAP*.

Martinez, A.M., and A.C. Kak (2001). PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **23**(2), 228–233.

Messer, K., J. Matas, J. Kittler, J. Luettin, and G. Maitre (1999). XM2VTSDB: the extended M2VTS database. In *Proceedings of AVBPA'99*, Washington D.C. pp. 72–77.

Mihelič F. and J. Žibert (2006). Robust speech detection based on phoneme recognition features. *Lecture Notes in Artifical Intelligence*, **4188**, 755–462.

Nicolai, B.M., K.I. Theron and J. Lammertyn (2007). Kernel PLS regression on wavelet transformed NIR spectra for prediction of sugar content of apple. *Chemometrics and Intelligent Laboratory Systems*, **85**(2), 243–252.

Ribarić, S., I. Fratrić and K. Kiš (2008). A novel biometric personal verification system based on the combination of palmprints and faces. *Informatica*, **19**(1), 81–100.

Rosipal, R. (2003). Kernel partial least squares for nonlinear regression and discrimination. *Neural Ntwork World*, **13**(3), 291–300.

Rosipal, R., and L.J. Trejo (2001). Kernel partial least squares regression in reproducing kernel Hilbert space. *Journal of Machine Learning Research*, **2**, 97–123.

Samaria, F., and A. Harter (1994). Parameterisation of a stochastic model for human face identification. In *2nd IEEE Workshop on Applications of Computer Vision*, Sarasota (Florida). pp. 138–142.

Schölkopf, B., A.J. Smola and K.R. Müller (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, **10**, 1299–1319.

Shen, L., and L. Bai (2006). MutualBoost learning for selecting Gabor features for face recognition. *Pattern Recognition Letters*, **27**(15), 1758–1767.

Shen, L., L. Bai and M. Fairhurst (2007). Gabor wavelets and general discriminant analysis for face identification and verification. *Image and Vision Computing*, **25**(5), 553–563.

Štruc, V., F. Mihelič and N. Pavešić (2008). Face authentication using a hybrid approach. *Journal of Electronic Imaging*, **17**(1), 011003.

Su, C.L. (2007). Overlapped finger geometry signal processing and finger shape comparisons for person identification. *Informatica*, **18**(3), 447–456.

Turk, M., and A. Pentland (1991). Eigenfaces for recognition. *Journal of Cognitive Neurosicence*, **3**(1), 71–86.

Wiskott, L., J.M. Fellous, N. Kruger, C. von der Malsburg (1997). Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**, 775–779.

Wold, H. (1966). Estimation of principal components and related models by iterative least squares. In P.R. Krishnaiah (Ed.), *Multivariate Analysis*. Academic Press, New York. pp. 391–420.

Yang, M.H. (2002). Kernel eigenfaces vs. kernel fisherfaces: face recognition using kernel methods. In *Proceedings of the 5th IEEE ICAFGR*, Washington D.C. pp. 215–220.

Yang, C., R. Duraiswami and L. Davis (2004). Efficient kernel machines using the improved fast Gauss transform. *Advances in Neural Information Processing Systems*, **16**, 1561–1568.

Žibert, J., and F. Mihelič (2002). Zhao–Atlas–Marks representation of speech signals. *Electrotechnical Review*, **69**(3–4), 159–164.

V. Štruc was born in 1981. He received his BSc degree in electrical engineering from the University of Ljubljana in 2005. He is currently working as a researcher at the Laboratory of Artificial Perception, Systems and Cybernetics at the Faculty of Electrical Engineering of the University in Ljubljana. His research interests include pattern recognition, machine learning and biometrics.

N. Pavešić was born in 1946. He received his BSc degree in electronics, M.Sc. degree in automatics, and PhD degree in electrical engineering from the University of Ljubljana, Slovenia, in 1970, 1973 and 1976, respectively. Since 1970 he has been a staff member at the Faculty of Electrical Engineering in Ljubljana, where he is currently head of the Laboratory of Artificial Perception, Systems and Cybernetics. His research interests include pattern recognition, neural networks, image processing, speech processing, and information theory. He is the author and co-author of more than 100 papers and 3 books addressing several aspects of the above areas.

Gabor bangelėmis pagrįsti branduolio dalinių mažiausiųjų kvadratų atskyrimo bruožai veidams atpažinti

Vitomir ŠTRUC, Nikola PAVEŠIĆ

Straipsnis siūlo naują metodą veido bruožams išskirti, pagrįstą veido vaizdų atvaizdavimu Gabor bangelėmis ir branduolio dalinių mažiausiųjų kvadratų atskyrimo algoritmu. Pasiūlytas metodas bruožams išskirti atliekamas dviem nuosekliais žingsniais. Pirmajame žingsnyje keturiasdešimties Gabor bangelių aibė naudojama išskirti skiriamiesiems patikimiems veido bruožams, o antrajame – branduolio dalinių mažiausiųjų kvadratų atskyrimas naudojamas Gabor bruožų vektoriaus matmenų skaičiui sumažinti ir skiriamumui padidinti. Eksperimento rezultatai rodo, kad pasiūlytas būdas yra efektyvesnis už pagrindinių komponenčių analizę, tiesinę diskriminantinę analizę, branduolio pagrindinių komponenčių analizę, apibendrintą diskriminantinę analizę ir jų kombinacijas su veido vaizdų Gabor atvaizdavimu.