Face authentication using a hybrid approach

V. Štruc, F. Mihelič and N. Pavešić

Faculty of Electrical Engineering, University of Ljubljana,
Tržaška 25, SI-1000 Ljubljana, Slovenia

Abstract. This paper presents a hybrid approach to face-feature extraction based on the trace transform and the novel kernel partial-least-squares discriminant analysis (KPA). The hybrid approach, called trace kernel partial-least-squares discriminant analysis (TKPA) first uses a set of fifteen trace functionals to derive robust and discriminative facial features and then applies the KPA method to reduce their dimensionality. The feasibility of the proposed approach was successfully tested on the XM2VTS database, where a false rejection rate (FRR) of 1.25% and a false acceptance rate (FAR) of 2.11% were achieved in our best-performing face-authentication experiment. The experimental results also show that the proposed approach can outperform kernel methods such as generalized discriminant analysis (GDA), kernel fisher analysis (KFA) and complete kernel fisher discriminant analysis (CKFA) as well as combinations of these methods with features extracted using the trace transform.

Keywords: Face recognition, Trace transform, Kernel partial-least-squares discriminant analysis, XM2VTS database

Author information: (Send correspondence to V. Š.)
V.Š.: E-mail: vitomir.struc@fe.uni-lj.si, Telephone: +386 1 4768 839, Fax: +386 1 4768 316
F.M.: E-mail: mihelicf@fe.uni-lj.si, Telephone: +386 1 4768 841, Fax: +386 1 4768 316
N.P.: E-mail: nikolap@fe.uni-lj.si, Telephone: +386 1 4768 315, Fax: +386 1 4768 316

1 INTRODUCTION

In recent decades automatic face recognition has become one of the most active research areas in the field of biometrics, with possible applications in law enforcement, access control, e-banking, entertainment, smart homes, and numerous other domains [1, 2].

Based on the face representation used for the recognition, face-recognition techniques can be divided into two groups [3]: *appearance-based* and *feature-based*. Appearance-based methods consider the global properties of the face and use the whole face image (or specific image regions) to extract facial features, while feature-based methods, on the other hand, rely on local facial characteristics (such as eyes, nose and mouth) and use angles and distances between fiducial points on the face as descriptors for face recognition.

A typical example of appearance-based methods is the principal component analysis (PCA), introduced by Turk and Pentland [4]. PCA identifies a subspace whose basis vectors correspond to the maximum variance directions of the given training images. Face recognition is performed by comparing the projection coefficients of different face images in this subspace, in which information compression, dimensionality reduction and decorrelation of the pixel values of the face images are achieved.

Another representative of appearance-based methods is linear discriminant analysis (LDA), which tries to improve upon PCA by considering the class-membership information of the training images. Unlike PCA, LDA constructs a linear subspace by maximizing the ratio of the between-class scatter matrix to the within-class scatter matrix, which serves as a criterion for class separability. Several techniques have been presented in the literature that make use of this discrimination criterion, including the Fisherface technique [5], Regularized LDA (RLDA) [6], Enhanced LDA (ELDA) [7], Fractional-step LDA (F-LFA) [8] and Direct LDA (DLDA) [9].

To cope with nonlinear changes in the appearance of the face images due to varying illumination, pose and facial expression, linear appearance methods (such as PCA or LDA) were recently extended to their nonlinear (kernel) form. Popular kernel appearance-based methods are kernel principal component analysis (KPCA) [10], generalized discriminant analysis (GDA) [11], kernel Fisher analysis (KFA) [12], complete kernel Fisher discriminant analysis (CKFA) [13] and the kernel fractional-step discriminant analysis (KFDA) [14].

Srisuk et al. [15] proposed a face-recognition technique based on the trace transform [16]; this can also be considered to be appearance-based as features are extracted from the whole face image. The proposed technique first extracts shape information from binarized trace transforms of a face image and then uses a dissimilarity measure, called the Hausdorff context, for the recognition.

Although appearance-based methods have been dominant face-recognition techniques for years, it is widely believed [17] that local face-features are more robust to face-image changes caused by varying illumination, pose and facial expression and are therefore very suitable for face recognition.

Examples of feature-extraction techniques that make use of the local characteristics of face images can be found in Gabor wavelet-based methods. One of the first methods to incorporate Gabor-based features was the dynamic link architecture (DLA) presented in [18]. In DLA a rectangular graph is placed over a 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 the recognition. Based on DLA, Wiskott et al. [19] 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).

Liu and Wechsler combined elements from appearance-based and feature-based methods and proposed the Gabor-Fisher Classifier (GFC) [20]. The GFC first employs a set of forty Gabor wavelets to derive an augmented Gabor-feature vector and then uses the enhanced Fisher linear discriminant model (EFM) to reduce the vector's dimensionality. Several modifications of the described approach were presented in the literature, including [1, 12, 21].

This paper introduces a hybrid feature-extraction approach, called trace kernel partial-least-squares discriminant analysis (TKPA), which applies the novel kernel partial-least-squares discriminant analysis (KPA) to a feature vector derived by using the trace transform. Experimental results on the XM2VTS database show that the trace face-feature vector can significantly improve the authentication performance of the subspace-projection techniques and that the proposed approach (TKPA) can outperform popular face-recognition techniques such as GDA, KDA and CKFA as well as combinations of these techniques with features extracted by the trace transform. The novelty of this paper therefore comes from:

- the derivation of the trace face-feature vector,
- the kernel partial-least-squares discriminant analysis (KPA), which improves upon Yang's CKFA [13] by replacing the kernel principal component analysis (KPCA) subspace with the kernel partial-least-squares (KPLS) subspace,
- the integration of the trace transform and the KPA technique into the hybrid TKPA feature-extraction approach,
- a comparative assessment of the proposed hybrid approach and several state-of-the-art feature-extraction methods.

The rest of the paper is organized a follows. In Section 2 the theory behind the trace transform is reviewed and the methodology to derive the trace face-feature vector is outlined. The kernel partial-least-squares discriminant analysis is introduced in Section 3 and the novel TKPA feature-extraction approach is detailed in Section 4. Section 5 describes the matching procedure, while the experimental results are given in Section 6. The paper concludes with some final comments and directions for future work in Section 7.

2 THE TRACE FACE-FEATURE VECTOR

The trace transform (TT), first introduced by Kadyrov and Petrou in [16], is a generalization of the Radon transform, which can be used for the extraction of robust image features and subsequently for object recognition. It has already been employed in the field of face recognition (i.e., face authentication), yielding excellent results [15, 22], but its wider use is still limited due to the complexity of the algorithms used for TT-based feature extraction. This section introduces a novel, simplified, approach for the derivation of the TT-based face-feature vector, which in contrast to the methods presented in [15, 22] only relies on a pair of eye coordinates and thus eliminates the need for any masking of the facial region, threshold selection, weight calculation, etc. As will be shown in Section 6, the proposed trace face-feature vector (in comparison with intensity images) improves the performance of different subspace projection techniques by at least 30%, making the trace transform a useful image preprocessing step for future biometric systems.

2.1 The trace transform

Let $I(x, y) \in \mathbb{R}^{a \times b}$ represent an image of size $a \times b$ (in pixels) and let $\mathcal{L}$ denote a set of all the possible lines (called tracing lines) through $I(x, y)$, determined by the parameters p and θ. The parameter p denotes the shortest distance from a tracing line to the center of the image O_c, while θ defines the angle of the tracing lines normal to the reference direction (i.e., the x-axis). We further assume that t represents a variable defining the position on a tracing line (see Fig. 1 for a visualization of these parameters). The trace transform (TT) [15, 16, 22] can then be defined as a function $w(p, \theta)$, which is calculated by applying the trace functional F on all the lines of $\mathcal{L}$, i.e.:

$$w(p, \theta) = F(g(\theta, p, t)) \tag{1}$$

where $g(\theta, p, t)$ denotes the function values of the tracing line defined by the pair (p, θ), and the parameters θ and p lie in the range $[0, 2\pi]$ and $[-p_d, p_d]$, respectively. Note that p_d equals half the length of the diagonal of $I(x, y)$.

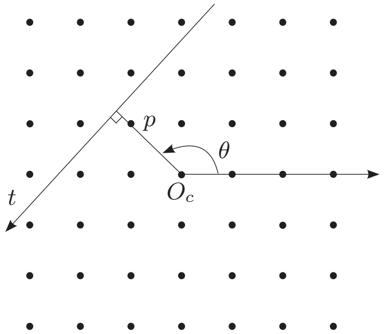


Fig. 1. Example of a line with parameters p and θ through a sample image.

With a closer examination of the equation (1) we can see that the TT provides a mapping from the original image space to a 2D parameter space spanned by p and θ. A trace-transformed image can therefore again be interpreted as an image (i.e., $w(p, \theta)$) whose pixel locations are given by p and θ and pixel values by the values of the functional F on the corresponding tracing lines. Using different trace functional results in different TTs, which depending on the functional used exhibit different properties that can be useful for face recognition.

In our experiments we made use of the 15 trace functionals that are presented in Table 1. Note that there is no specific rule about which or how many functionals should be used for the calculation of the trace transform. For example, Srisuk et al. [22] made use of 22 functionals to derive their features with the trace transform. In this paper we selected a set of 15 functionals,

Number	Trace functional	Number	Trace functional
1	$F_1(f(t)) = \int_0^\infty f(t)dt$	9	$F_9(f(t)) = median\{f(t)'\}$
2	$F_2(f(t)) = (\int_0^\infty \|f(t)\|^{0.5}dt)^2$	10	$F_{10}(f(t)) = max\{f(t)'\}$
3	$F_3(f(t)) = median\{f(t)\}$	11	$F_{11}(f(t)) = range\{f(t)'\}$
4	$F_4(f(t)) = max\{f(t)\}$	12	$F_{12}(f(t)) = var\{f(t)'\}$
5	$F_5(f(t)) = range\{f(t)\}$	13	$F_{13}(f(t)) = mad\{f(t)'\}$
6	$F_6(f(t)) = \int_0^\infty f(t)'dt$	14	$F_{14}(f(t)) = (\int_0^\infty \|f(t)'\|^{0.5}dt)^2$
7	$F_7(f(t)) = var\{f(t)\}$	15	$F_{15}(f(t)) = (\int_0^\infty \|f(t)'\|^2dt)^{0.5}$
8	$F_8(f(t)) = mad\{f(t)\}$		

Table 1. The trace functionals used for the derivation of the trace face-feature vector (max = maximum value, min = minimum value, $range$ = maximum value - minimum value, var = variance, mad = mean absolute deviation and $f(t) = g(\theta, p, t)$)

consisting mainly of statistical descriptors of the tracing lines (and their derivatives) to analyze the statistical properties of different face images and to derive the trace face-feature vectors.

2.2 Calculating the trace face-feature vector

Consider the set of 15 trace functionals $\mathcal{T} = \{F_i : i = 1, 2, ...15\}$ presented in Table 1. To derive the trace face-feature vector of a given face image $I(x, y)$, the image is first mapped into the trace space using all the functionals of $\mathcal{T}$. The mapping results in a set of 15 trace-transformed images $w_i(p, \theta)$ ($i = 1, 2, ..., 15$), each carrying information about the statistical properties of the tracing lines of the original input image. To incorporate all of this information in a single feature vector we concatenate all the trace-transformed images and derive the trace face-feature vector **x**. Before the concatenation, we first downsample the TTs $w_i(p, \theta)$ ($i = 1, 2, ..., 15$) to a size of $u \times v$ (where $u < a$ and $v < b$) to reduce the trace-space dimension, and then normalize them to zero mean and unit variance. In the last step we create vectors $\mathbf{w}_i$ out of the downsampled images $w_i(p, \theta)$ by concatenating their rows (or columns) and combine all of the vectors $\mathbf{w}_i$ ($i = 1, 2, ..., 15$) into the trace-face feature vector **x**, i.e.:

$$\mathbf{x} = (\mathbf{w}_1^T \;\; \mathbf{w}_2^T \;\; \cdots \;\; \mathbf{w}_{15}^T)^T \in \mathbb{R}^N \tag{2}$$

where T denotes the transpose operator and $N = 15uv$ (u and v denote the dimensions of the resized images $w_i(p, \theta)$ for $i = 1, 2, ..., 15$).

Three examples of the trace face-feature vector (the TTs are presented in image form) are shown in Fig. 2. The first two belong to the same person (Fig. 2 a) and 2 b)), while the third trace face-feature vector corresponds to the face image of a different person (Fig. 2 c)). It will be shown in Section 6.5 that the trace transform emphasizes the discriminative information contained in the original (face) images, leading to a reduction in the face-authentication error rates.

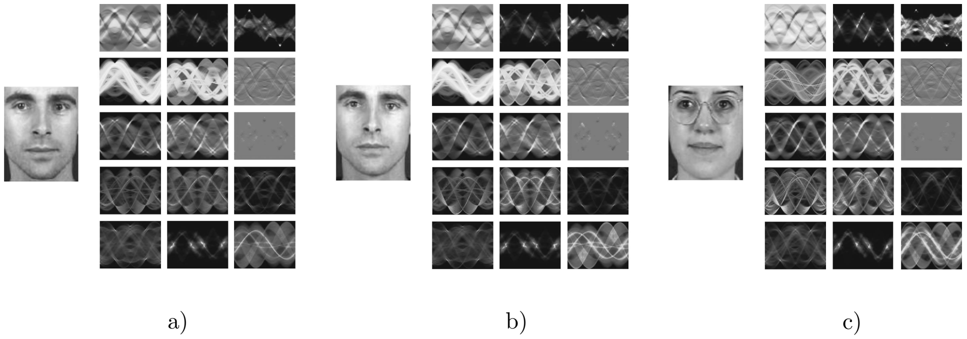


Fig. 2. Examples of trace-transformed images for a), b) the same person, c) a different person.

Although the procedure for deriving the trace face-feature vector is relatively simple in theory, there are several practical issues one has to consider:

- To calculate the trace transform of a face image, the parameters θ and p have to be quantized (e.g., in our experiments (Section 6) we used 180 values for the parameter θ taken from the range $[0, 2\pi]$ with a step size of $\Delta\theta = pi/90$ (i.e., $\theta = i\Delta\theta$, for $i = 1, 2, ...180$) and similarly, we used 140 discrete values (where 140 equals the length of the diagonal of a 88×110 image rounded up to an integer value) for the parameter p taken from the range $[-70, 70]$ with a step size of $\Delta p = 1$; thus, the size of one TT image equaled 140×180).

- To reduce the size of the final trace face-feature vector, each of the fifteen TT images $w_i(p,\theta)$ ($i = 1, 2, ..., 15$) has to be downsampled prior to concatenation (e.g., we used a rectangular sampling grid with 35 horizontal and 45 vertical grid lines uniformly distributed over the trace-transformed image to reduce the size of the $w_i(p,\theta)$ ($i = 1, 2, ..., 15$) from 140×180 to 35×45).

Despite the downsampling of the TTs to a size of 35×45, the concatenation of all fifteen trace-transformed images still results in a feature vector whose dimension is $N = 23,625$. To further reduce the vector's dimension we will subject it to a novel subspace-projection technique, which will be presented in the next section.

3 KERNEL PARTIAL-LEAST-SQUARES DISCRIMINANT ANALYSIS (KPA)

Subspace projection techniques based on Fisher's criterion for class separability are among the most popular methods for facial feature extraction. Numerous discriminant approaches have been presented in the literature in the past, including the famous Fisherface approach [5], regularized linear discriminant analysis (RLDA) [6] and direct linear discriminant analysis (DLDA) [9]. In recent years, however, kernel variants of these methods have emerged as a powerful tool for nonlinear feature extraction. Popular examples of kernel discriminant methods are the generalized discriminant analysis (GDA) [11], kernel Fisher analysis (KFA) [12] and the complete kernel Fisher discriminant analysis (CKFA) [13]. Following the idea of CKFA (where features are extracted by first applying kernel principal component analysis (KPCA) to reduce the space dimension and subsequently using LDA in the reduced space) we propose a novel kernel discriminant approach, called Kernel partial-least-squares discriminant Analysis (KPA), where the KPCA part of the CKFA is replaced with the kernel partial-least-squares (KPLS) algorithm.

In the remainder of this section we first review the basic concept of the partial-least-squares algorithm, then a description of KPLS is given, and finally the KPA method is introduced.

3.1 Extracting latent vectors with kernel partial least squares (KPLS)

The kernel partial-least-squares (KPLS) algorithm [23] is a nonlinear extension of the partial-least-squares technique presented in [24]. It can be used for relating several dependent variables (responses) to a large number of independent variables (predictors). The basic idea of KPLS is to create latent vectors (components, factors) that account for as much of the variance of the independent variables as possible, while simultaneously modeling the dependent variables well.

3.1.1 The concept of partial least squares (PLS)

Denote by $\mathbf{X} = \left[\mathbf{x}_1^T, \mathbf{x}_2^T, ..., \mathbf{x}_n^T\right]^T$ a $n \times N$ matrix of n N-dimensional input variables $\mathbf{x}$ and by $\mathbf{Y}$ a $n \times C$ matrix of corresponding n C-dimensional response variables $\mathbf{y}$. The PLS algorithm finds the matrices $\mathbf{T}$, $\mathbf{P}$, $\mathbf{E}$, $\mathbf{U}$, $\mathbf{Q}$ and $\mathbf{F}$ in a way that the matrices $\mathbf{X}$ and $\mathbf{Y}$ can be expressed as follows [25]:

$$\begin{aligned} \mathbf{X} &= \mathbf{T}\mathbf{P}^T + \mathbf{E} \\ \mathbf{Y} &= \mathbf{U}\mathbf{Q}^T + \mathbf{F} \end{aligned} \tag{3}$$

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

The matrices of the decomposition in (3) are computed with the help of the nonlinear iterative partial-least-squares (NIPALS) algorithm [24] (the NIPALS algorithm for the KPLS method will be described in more detail in the next section), which finds the weight matrices **P** and **Q** in such a way that the covariance between the **X** and **Y** matrices is maximized.

3.1.2 The kernel partial least squares (KPLS) algorithm

To extend the PLS algorithm to its nonlinear form (kernel partial least squares - KPLS), Rosipal and Trejo [23] proposed a modification of the NIPALS algorithm based on the ”kernel trick” [10], which avoids calculating the potentially computationally expensive nonlinear transformation Φ (where Φ denotes a nonlinear mapping of the vector $\mathbf{x}$ from the input space $\mathbb{R}^N$ to a high dimensional feature space $\mathcal{F}$, i.e.: $\Phi : \mathbf{x} \in \mathbb{R}^N \rightarrow \Phi(\mathbf{x}) \in \mathcal{F}$), and finds the parameters of the decomposition (3) using the $n \times n$ kernel matrix of the input variables $\mathbf{x}_i$ $(i = 1, 2, ..., n)$:

$$\mathbf{K} = \left[\Phi(\mathbf{x}_i)\Phi(\mathbf{x}_j)^T\right] = \left[K(\mathbf{x}_i, \mathbf{x}_j)\right]; \; \forall i, j \tag{4}$$

As our goal is to compute latent vectors (i.e., the matrices **T** and **U**) that are useful for classification, we construct the $n \times C$ response matrix **Y** in the following way [25–27]:

$$\mathbf{Y} = \begin{bmatrix} \mathbf{1}_{n_1} & \mathbf{0}_{n_1} & \cdots & \mathbf{0}_{n_1} \\ \mathbf{0}_{n_2} & \mathbf{1}_{n_2} & \cdots & \mathbf{0}_{n_2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{n_C} & \mathbf{0}_{n_C} & \cdots & \mathbf{1}_{n_C} \end{bmatrix}, \tag{5}$$

where C denotes the number of classes in the set of n N-dimensional inputs (matrix **X**), n_i denotes the number of inputs in class C_i, $\mathbf{1}_{n_i}$ $(i = 1, 2, ..., C)$ stands for a $n_i \times 1$ vector of all ones, and $\mathbf{0}_{n_i}$ $(i = 1, 2, ..., C)$ is a $n_i \times 1$ vector of all zeros. It is easy to see that the rows of **Y** represent encoded labels of the C classes of **X**.

The NIPALS algorithm for the KPLS approach can now be described as follows [28]:

Input: Matrices $\mathbf{K}_c$ and **Y**

Output: r n-dimensional latent vectors **t** and **u**

1. randomly initialize the **y**-latent vector **u**
2. calculate the **x**-latent vector:

$$\mathbf{t} = \mathbf{K}_c\mathbf{K}_c^T\mathbf{u}, \;\; \mathbf{t} \leftarrow \mathbf{t}/\|\mathbf{t}\|$$

3. update the **y**-latent vector:

$$\mathbf{u} = \mathbf{K}_c\mathbf{K}_c^T\mathbf{t}, \;\; \mathbf{u} \leftarrow \mathbf{u}/\|\mathbf{u}\|$$

4. repeat the steps 2-3 until convergence (i.e., the vector **u** in two consecutive iterations does not change)
5. deflate the matrices $\mathbf{K}_c$ and **Y**:

$$\mathbf{K}_c = \mathbf{K}_c - \mathbf{t}\mathbf{t}^T\mathbf{K}_c,\ \ \mathbf{Y} = \mathbf{Y} - \mathbf{t}\mathbf{t}^T\mathbf{Y}$$

6. orthonormalize the matrix **Y**:

$$\mathbf{Y} = \mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1/2}$$

7. continue with step 2 using new matrices $\mathbf{K}_c$ and **Y**.

Note that in the above algorithm the vectors **t** and **u** denote columns of the matrices **T** and **U**, respectively, while $\mathbf{K}_c$ denotes the centered kernel matrix of the input variables in **X**, i.e. [10]:

$$\mathbf{K}_c = (\mathbf{I} - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T)\mathbf{K}(\mathbf{I} - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T), \tag{6}$$

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

After the matrices **T** and **U** are calculated, the latent vectors of new inputs $\mathbf{X}^*$ are computed from:

$$\mathbf{T}^* = \mathbf{K}_c^*\mathbf{A} \tag{7}$$

where **A** is the $n \times r$ KPLS projection matrix, defined by

$$\mathbf{A} = \mathbf{K}_c^T\mathbf{U}(\mathbf{T}^T\mathbf{K}_c\mathbf{K}_c^T\mathbf{U})^{-1} \tag{8}$$

and

$$\mathbf{K}_c^* = (\mathbf{K}^* - \frac{1}{n}\mathbf{1}_{n^*}\mathbf{1}_n^T\mathbf{K})(\mathbf{I} - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T) \tag{9}$$

is the centered kernel matrix of the new inputs in $\mathbf{X}^*$ (n^* denotes the number of new inputs).

In this paper four polynomial kernels are considered for the implementation of the KPLS algorithm, i.e., the linear or first-order polynomial kernel, the second-order , the third-order and the forth-order polynomial kernel, i.e.:

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T\mathbf{x}_j)^z \tag{10}$$

where $z = 1, 2, 3$ and 4.

3.2 Extracting features with KPA

To extract the final face features with the novel kernel partial-least-squares discriminant analysis (KPA) we construct the within-class and between-class scatter matrices (denoted by $\mathbf{S}_{w_{lat}}$ and $\mathbf{S}_{b_{lat}}$) using the first $r = n - C$ (recall that r denotes the number of extracted latent vectors, n denotes the number of input variables in **X**, and C stands for the number of classes in **X**) latent vectors of the input variables in **X**.

Let **T** represent a $n \times r$ matrix, containing in its columns $r = n - C$ latent vectors **t**; further denote the n row vectors of **T** by **s**. The matrices $\mathbf{S}_{w_{lat}}$ and $\mathbf{S}_{b_{lat}}$ can then be defined as follows [5]:

$$\mathbf{S}_{w_{lat}} = \sum_{i=1}^{C}\sum_{\mathbf{s}_j \in C_i}(\mathbf{s}_j - \bar{\mathbf{s}}_i)(\mathbf{s}_j - \bar{\mathbf{s}}_i)^T \tag{11}$$

$$\mathbf{S}_{b_{lat}} = \sum_{i=1}^{C}|C_i|(\bar{\mathbf{s}}_i - \bar{\mathbf{s}})(\bar{\mathbf{s}}_i - \bar{\mathbf{s}})^T \tag{12}$$

where $\mathbf{s}_j$ represents the j-th row vector of $\mathbf{T}$ corresponding to the j-th input variable of $\mathbf{X}$, $\bar{\mathbf{s}}_i$ represents the mean vector of the rows of $\mathbf{T}$ belonging to class i, $\bar{\mathbf{s}}$ is the mean vector of all the rows of $\mathbf{T}$, and $|C_i|$ stands for the number of variables in class i ($i = 1, 2, ..., C$).

Based on these matrices, the KPA method calculates a projection matrix $\mathbf{Z}$ that maximizes Fisher's criterion $J(\mathbf{Z})$, i.e:

$$J(\mathbf{Z}) = \frac{\mathbf{Z}^T \mathbf{S}_{b_{lat}} \mathbf{Z}}{\mathbf{Z}^T \mathbf{S}_{w_{lat}} \mathbf{Z}}, \ \ (\mathbf{Z} \neq \mathbf{0}) \tag{13}$$

Since the within-class scatter matrix was computed using only the first $n - C$ latent vectors, $\mathbf{S}_{w_{lat}}$ is always nonsingular and the optimal projection matrix $\mathbf{Z}$ can be constructed by concatenating the eigenvectors of the generalized eigenproblem defined by:

$$\mathbf{S}_{b_{lat}} \mathbf{z}_i = \lambda_i \mathbf{S}_{w_{lat}} \mathbf{z}_i \tag{14}$$

where $\mathbf{z}_i$ denotes the i-th eigenvector of (14) and λ_i represents the corresponding i-th eigenvalue.

The $(n - C) \times d$ projection matrix $\mathbf{Z}$ is thus defined as follows:

$$\mathbf{Z} = [\mathbf{z}_1 \ \ \mathbf{z}_2 \ \ \cdots \ \ \mathbf{z}_d] \tag{15}$$

where d is the number of eigenvectors that defines the dimensionality of the KPA subspace ($d \leq C - 1$) and the eigenvectors $\mathbf{z}_i$ ($i = 1, 2, ..., d$) correspond to a set of decreasing eigenvalues, i.e., $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$.

Using the projection matrix $\mathbf{Z}$ (15), the final feature vector $\mathbf{y}$ can be computed from the vector $\mathbf{s}$ of a given input variable $\mathbf{x}$, i.e.:

$$\mathbf{y} = \mathbf{s}\mathbf{Z} \tag{16}$$

where the feature vector $\mathbf{y}$ is of size $1 \times d$.

4 TRACE KERNEL PARTIAL-LEAST-SQUARES DISCRIMINANT ANALYSIS (TKPA)

In this **section** we present a novel, hybrid feature-extraction approach which combines the trace face-feature vector, introduced in Section 2, with the novel KPA subspace projection technique presented in the previous section. The proposed approach is based on the following facts:

- the trace transform can extract features that are robust against smaller changes in pose, illumination and facial expression [22] (as the trace-transformed images are calculated by applying different functionals to the *tracing lines* of a face image, they are to some extent insensitive to changes of the individual pixel values of the input face image);

- the trace face-feature vector contains more discriminative information than the original grey-scale input image and is therefore better suited for use with subspace methods (note that the trace transform with the first functional of $\mathcal{T}$ represents the Radon transform, which is known to contain the same information as the input image [22,29]; the remaining 14 functionals of $\mathcal{T}$ emphasize different characteristics of the input image and therefore provide supplementary information, useful for face recognition);

- it was shown in [26] that LDA-based features, computed in the PLS subspace (which is constructed using the class membership information of the training images), ensure better authentication performance than the Fisherface method [5], where discriminative features are computed in the PCA subspace (this also applies to KPCA and KPLS).

Based on these facts the hybrid approach, called the trace kernel partial-least-squares discriminant analysis or TKPA, first computes the trace face-feature vector **x** from an input face image $I(x, y)$ following the procedure described in Section 2 and then maps the vector **x** into the low-dimensional KPA subspace, where the final face authentication is performed.

The whole TKPA algorithm can be described as follows:

Training

Input: A set of n training images of the given database $\mathcal{X}$

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

1. compute the trace transforms of all the training images in $\mathcal{X}$ using the set of 15 trace functionals $\mathcal{T}$ (1),
2. resize the trace-transformed images $w(p, \theta)$ to reduce the trace feature-space dimension,
3. normalize the resized images $w(p, \theta)$ to zero mean and unit variance,
4. combine the normalized and resized trace-transformed images of $\mathcal{X}$ into n trace face-feature vectors **x** (2),
5. construct matrices **K** (4), $\mathbf{K}_c$ (6) and **Y**, (5) using all the trace face-feature vectors of $\mathcal{X}$
6. compute $r = n - C$ latent vectors **t** using the KPLS NIPALS algorithm,
7. construct the KPA projection matrix **Z** (15),
8. compute the low-dimensional KPA feature vector $\mathbf{y}_i$ ($i = 1, 2, ..., n$) for all the training images in $\mathcal{X}$,
9. calculate the client models (the mean feature vectors of all the training images belonging to a specific client) for all C clients of the given database.

Feature extraction for new (test) images

Input: A set of n^* test images of the given database $\mathcal{S}$

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

1. follow the steps 1-4 of the TKPA training algorithm using a set of test images $\mathcal{S}$,
2. compute the matrices $\mathbf{K}^*$ (4) and $\mathbf{K}_c^*$ (9) using the trace face-feature vectors from $\mathcal{S}$ and $\mathcal{X}$,
3. compute the $r = n - C$ latent vectors **t** for each trace face-feature vector of $\mathcal{S}$ (7),
4. compute the low-dimensional KPA feature vector $\mathbf{y}_i$ ($i = 1, 2, ..., n^*$) for all n^*test images in $\mathcal{S}$.

5 MATCHING

In the matching stage the feature vector **y** (16) of a given input face image (sometimes referred to as the live feature vector) is compared with the claimed client's feature vector (i.e., the mean feature vector of the clients training images). If the live feature vector and the claimed client's feature vector display a degree of similarity that is higher than the decision threshold, then the input image is recognized as genuine (i.e., belonging to the claimed client), otherwise the input image is recognized as belonging to an impostor (or vice versa, if a dissimilarity measure is used).

In this paper the gradient distance d_{gd} [30] was used as the similarity measure. The d_{gd}-based matching score was computed as follows:

$$d_{gd}(\mathbf{y}, \bar{\mathbf{y}}_i) = \frac{\|(\mathbf{y} - \bar{\mathbf{y}}_i)^T \nabla P(i|\mathbf{y})\|}{\|\nabla P(i|\mathbf{y})\|} \tag{17}$$

where $\nabla P(i|\mathbf{y})$ is defined as

$$\nabla P(i|\mathbf{y}) = \sum_{j=1,\ \ j\neq i}^{C} p(\mathbf{y}|j)(\bar{\mathbf{y}}_j - \bar{\mathbf{y}}_i) \tag{18}$$

and $p(\mathbf{y}|j)$ equals

$$p(\mathbf{y}|j) = \frac{1}{\sqrt{(2\pi)^d}} exp\{-\frac{1}{2}(\mathbf{y} - \bar{\mathbf{y}}_j)^T(\mathbf{y} - \bar{\mathbf{y}}_j)\}. \tag{19}$$

In the above equations **y** denotes the feature vector of the given input image, $\bar{\mathbf{y}}_i$ and $\bar{\mathbf{y}}_j$ denote the mean feature vectors of the i-th and j-th client ($i, j = 1, 2, ..., C$), C denotes the total number of clients, d is the KPA subspace dimension,$\nabla P(i|\mathbf{y})$ stands for the gradient direction of the i-th client's aposteriori probability function, $p(\mathbf{y}|j)$ represents the j-th clients measurement distribution, and, finally, $\|.\|$ and T denote the norm and the transpose operator, respectively.

Note that the gradient distance was chosen as the similarity measure in this paper because it was specifically developed to be used in conjunction with feature-extraction methods that are based on Fisher's criterion (13) for class separability [30].

6 EXPERIMENTS AND DISCUSSION

6.1 Database and experimental protocol

Our experiments were conducted on face images from the multi-modal XM2VTS database [31], which contains audio and video data as well as color images of 295 individuals, recorded in four separate sessions that were uniformly distributed over a period of five months. At each session two recordings were made, resulting in a total of 2360 frontal face images (8 per subject) that were available for the training and testing of the proposed TKPA feature-extraction algorithm.

To enable a direct comparison of our approach with other methods presented in the literature the first configuration of the standard experimental protocol (i.e., the Lausanne protocol [32]), associated with the XM2VTS database, was used in the experiments. The protocol splits the 295 subjects into groups of 200 clients (i.e., $C = 200$) and 95 impostors (25 used for the evaluation and 70 for the testing). The images of these groups are then further divided into sets of training, evaluation and test images. Specifically, the first configuration of the Lausanne protocol resulted in the following experimental setup [32]:

- client training examples: 3 images per client
- client access trials in the evaluation stage: $q_{ce} = 600$
- impostor access trials in the evaluation stage: $q_{ie} = 40,000$ (25 × 8 × 200)

- client access trials in the test stage: $q_{ct} = 400$
- impostor access trials in the test stage: $q_{it} = 112,000$ ($70 \times 8 \times 200$)

The images of the training set were used for the calculation of the KPA face-feature vectors 16 and for the construction of the client models (i.e., the mean feature vectors). The images of the evaluation set were employed for the tuning of the KPA subspace projection technique (i.e., selection of the kernel function and the optimal number of features) and for an estimation of the global decision threshold that is used for accepting or rejecting a person claiming a client identity, while the images of the test set were employed only for the final performance evaluation.

Two standard error rates were used in our experiments for measuring the performance of the proposed TKPA approach. The first, called the false acceptance rate (FAR), is defined as the ratio of successful impostor accesses q_{si} to the number of impostor access trials q_i (i.e., q_{ie} in the evaluation or q_{it} in the test stage, respectively), while the second, called the false rejection rate (FRR), is defined as the ratio of unsuccessful client access trials q_{uc} to the total number of client access trials q_c (i.e., q_{ce} in the evaluation or q_{ct} in the test stage, respectively). Additionally, the total error rate (TER) was computed in each experiment to provide a single quantitative error measure of our approach. The described error rates are given by:

$$FAR = (\frac{q_{si}}{q_i})100\% \quad FRR = (\frac{q_{uc}}{q_c})100\% \tag{20}$$

$$TER = FAR + FRR \tag{21}$$

In addition to serving as quantitative performance measures for a biometric authentication system, the FAR and FRR are used to compute the decision threshold, which is usually set to a value that guaranties equal false acceptance and false rejection rates on the evaluation set.

6.2 Image preprocessing

Image preprocessing is one of the most important steps in (image-based) biometric authentication systems. It was shown in [33] that many face-recognition methods exhibit a drop in performance when images with poorly localized faces are used for the feature extraction instead of properly localized faces. Similarly, varying lighting conditions in the image-acquisition process also cause a deterioration of the authentication performance [34]. To avoid the presented problems we performed the following procedures:

- **Face localization** - for each face image of the XM2VTS database a pair of eye coordinates was manually determined.
- **Geometric normalization** - first, each image was rotated and scaled in such a way that the centers of the eyes were located at predefined positions; then the face parts of the image were cropped to a standard size of 88×110 pixels.
- **Photometric normalization** - each image was photometrically normalized using zero mean and unit variance normalization.

Some examples of the preprocessed face images from the XM2VTS database are shown in Fig. 3.

Fig. 3. Examples of preprocessed images from the XM2VTS database.

6.3 Kernel function selection

To select the most appropriate kernel function to be used in conjunction with the proposed TKPA feature-extraction approach, the first set of experiments evaluated the performance of the algorithm using different kernels. Only images from the evaluation set were used at this point, while the test images were reserved exclusively for the final comparative assessments presented in Sections 6.5 and 6.6. Four types of kernels were tested in the comparison, i.e., the first-, second-, third- and fourth-degree polynomial kernels (i.e., $K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i \cdot \mathbf{x}_j)^z$, for $z = 1, 2, 3$ and 4). The feature-vector length was set to its maximal value (i.e., $d = C - 1 = 199$) and the gradient distance (17) was used to calculate the matching scores.

The results of the described experiments are presented in Table 2. From the table we can see that the TKPA implementation with the third-degree polynomial ($z = 3$) kernel performed best, followed in order by the first-degree ($z = 1$) polynomial kernel, second-degree polynomial ($z = 2$) kernel and the fourth-degree polynomial ($z = 4$) kernel.

Poly. degree (z)	FAR(%)	FRR(%)	TER(%)
1	1.74	1.83	3.57
2	1.78	1.83	3.61
3	1.64	1.67	**3.41**
4	1.83	1.83	3.66

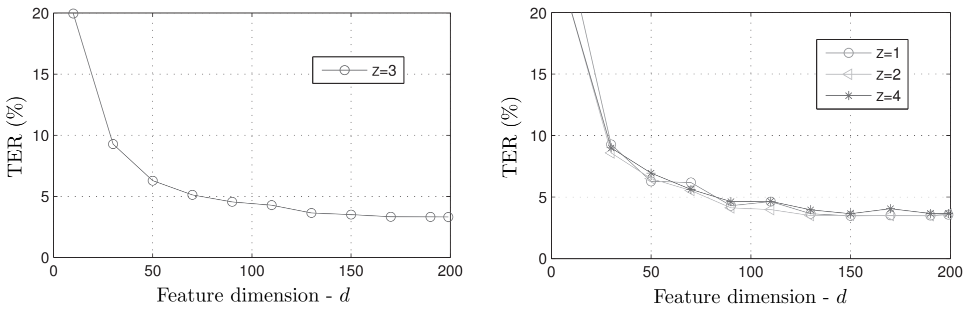
Table 2. Authentication performance of the TKPA with different kernel functions (evaluation set)

Based on the presented results we can conclude that the optimal kernel function to be used in conjunction with the TKPA approach is the third-degree polynomial kernel.

6.4 Selecting the optimal number of features

The second set of experiments assessed the performance of the proposed TKPA approach with respect to the dimension of the KPA subspace. For this purpose the dimension of the feature vector was gradually increased from $d = 10$ to its maximal value of $d = 199$ (recall that the maximum number of features equals $d = C-1 = 199$, where C denotes the number of clients in the database - Section 3.2) with a step size of $\Delta d = 20$ (except for the last step). Consistent with the findings of our previous experiments, the KPA part of the TKPA feature-extraction method was implemented using the third-degree polynomial kernel. Again, the gradient distance was used in the matching stage.

Fig. 4a shows the results of the assessment. From the graph (showing the dependency of the total error rate (TER) and the feature dimension d for the TKPA approach implemented with the third-degree polynomial kernel) we can see that the best authentication performance (on the evaluation set) was achieved when the maximum possible number of features was used (i.e.,



a) b)

Fig. 4. TKPA authentication performance with respect to the feature dimension (evaluation set), a) for the third-degree polynomial kernel, b) for the first-, second- and fourth-degree polynomial kernel

$d = 199$). In fact, each increase in the number of features led to an increase in the authentication performance of the TKPA approach. The presented findings suggest that the maximum number of features should be used for further comparative assessments of the proposed TKPA method.

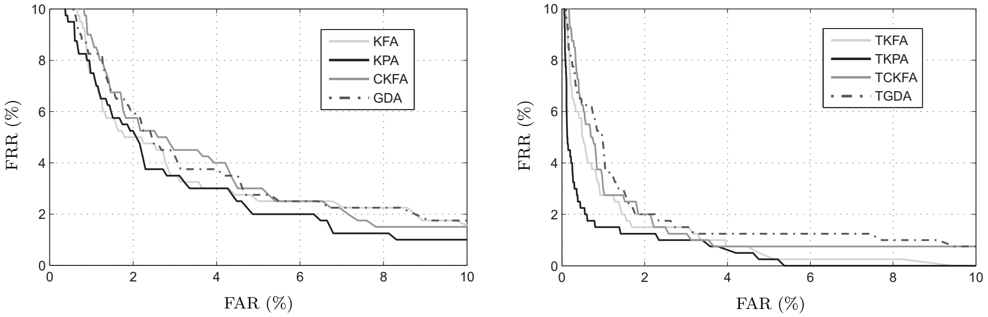
To validate our decision to use the third-degree polynomial kernel for the implementation of the TKPA approach, curves showing the dependency of TER and d for the remaining three kernel functions are shown in Fig. 4b. We can see that TKPA performs best (regardless of the kernel function) when all $d = 199$ features are used. The result supports our finding from the previous section that the optimal kernel function to be used in conjunction with the TKPA approach is the third-degree polynomial kernel.

6.5 Comparison with other methods

In **the third** set of experiments we compared the performance of the proposed TKPA feature-extraction approach to that of three state-of-the-art kernel methods that make use of Fisher's criterion (13) for deriving discriminative facial features. Specifically, the following methods were used for the comparison: GDA [11], KFA [12] and CKFA [13]. To assess the effect of the trace face-feature vector on the performance of the subspace projection techniques, all the listed methods were additionally implemented on the trace face-feature vectors (denoted by TGDA, TKFA and TCKFA).

The parameters (i.e., the number of features, the kernel function, the threshold, ...) of the feature-extraction methods used in the comparison were all optimized to yield the lowest possible error rates (on the evaluation set). The gradient distance was employed to calculate the matching scores of all the methods as it was specially designed to be used in conjunction with feature-extraction techniques based on Fisher's criterion (13) [30].

The ROC curves (displaying the dependency of the false acceptance rate and the false rejection rate at various operating points) of the authentication experiments are shown in Fig. 5, while the error rates for a specific threshold (i.e., the threshold that ensures equal error rates FAR and FRR on the evaluation set) are presented in Table 3. From the graphs and the table we can see that the KPA technique performs the best, both on the original grey-scale images (Fig. 5a) as well as on the trace face-feature vectors (Fig. 5b). If we compare all the tested methods we can further see that the top performer in the comparison was the TKPA method, followed in order by TKFA, TCKFA, TGDA, KPA, KFA, GDA and CKFA. Generally, all the methods performed significantly better when they were implemented on trace face-feature vectors instead of the original intensity images, achieving a performance improvement (of TER) of at least 40%.



a) b)

Fig. 5. ROC curves of the tested methods (test set), a) applied on grey-scale images, b) applied on trace face-feature vectors

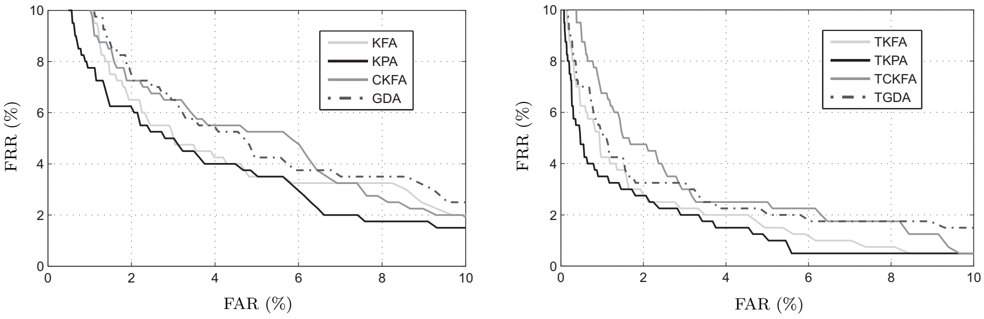
Method	FAR(%)	FRR(%)	TER(%)
GDA	4.06	3.75	7.81
KFA	3.76	3.00	6.67
CKFA	3.41	4.50	7.91
KPA	3.47	3.00	6.47
TGDA	2.44	1.75	4.19
TKFA	2.32	1.50	3.82
TCKFA	2.45	1.50	3.95
TKPA	2.11	1.25	**3.36**

Table 3. Comparison of the authentication performance for different methods (test set)

The reason for such a result is that trace-transformed images emphasize specific (depending on the functional used) information of the tracing lines of the face images, which seems to be of great importance for face recognition. Additionally, subspace methods performed on grey-scale pixel values exhibit great sensitivity to even small changes in pose, illumination or facial expression [34], while the TT, on the other hand, extracts features that are more robust with respect to the listed changes in the image characteristics [22].

6.6 Comparison with other methods using automatically registered images

Until now, all the described experiments were performed with manually registered images. However, it is well known that appearance-based methods (such as TKPA) are very sensitive to rotation and scale variations of the face images introduced by the face-localization procedure. Our last set of experiments therefore assessed the performance of the proposed TKPA (and KPA) approach and the six methods introduced in the previous section (i.e., GDA, KFA, CKFA, TGDA, TKFA and TCKFA) using automatically registered test images. The pair of eye coordinates needed for the extraction of the face region (see Section 6.2) was determined using the method proposed by Pozne, which located the eyes by maximizing a specially designed criterion function (see [35] for a detailed description of the method used). The localization procedure was applied to all the images of the client and impostor test sets and the resulting face images were then used for testing. The client models as well as the parameters (i.e, the kernel function, the feature vector length, the decision threshold, etc.) of the tested methods were determined using manually registered images, ensuring identical experimental conditions to those described in the previous sections (Sections 6.3-6.5).



a) b)

Fig. 6. ROC curves of the tested methods (using automatically registered test images), a) applied on grey-scale images, b) applied on trace face-feature vectors

Method	FAR(%)	FRR(%)	TER(%)
GDA	3.40	5.75	9.15
KFA	3.17	4.75	7.92
CKFA	3.73	5.75	9.48
KPA	3.13	4.75	7.88
TGDA	2.92	3.25	6.17
TKFA	2.06	2.75	4.81
TCKFA	2.85	3.25	6.10
TKPA	2.16	2.75	**4.66**

Table 4. Comparison of the authentication performance for different methods (using automatically registered test images)

A comparison of the ROC curves generated in our experiments is presented in Fig. 6. We can see that among all the tested methods the KPA technique was again found to perform the best, both in combination with the original intensity images (Fig. 6a) as well as in combination with the trace face-feature vectors (Fig. 6b). The same conclusion can also be made by considering the error rates presented in Table 4 (which shows the values of FAR, FRR and TER of the authentication experiments using a threshold that ensures equal FAR and FRR on the evaluation set).

Furthermore, we can see that the error rates obtained with automatically registered images (Table 4.) are considerably higher than those obtained with manually registered images (Table 3.) for all the tested methods, while the relative ranking in their performance remained the same. Nevertheless, all the subspace-projection techniques still performed significantly better when they were implemented on trace face-feature vectors rather than on the original grey-scale images (an improvement in the total error rate of more than 30% was observed for all the methods). As already pointed out in the previous section, there are two main reasons for such a result: first, the trace face-feature vectors contain more discriminative information than the original face images, and second, the trace transform produces face-features that are less sensitive to miss-registration than the intensity face images.

7 CONCLUSION AND FUTURE WORK

In this paper we have presented a novel trace kernel partial-least-squares discriminant Analysis (TKPA) algorithm for extracting features from frontal face images. The TKPA algorithm applies the KPA subspace-projection technique to a feature vector (i.e., the trace face-feature vector) which is constructed using the trace transform. The feasibility of the proposed approach was demonstrated in a series of authentication experiments, performed on the XM2VTS database. In a comparison where three popular discriminative kernel feature-extraction methods (i.e., GDA, KFA and CKFA) and combinations of these methods with trace face-feature vectors (i.e., TGDA, TKFA and TCKFA) were tested for their authentication performance in addition to the proposed TKPA approach, the TKPA performed best, achieving a total error rate of 3.36% on manually registered images and a total error rate of 4.66% on automatically registered images. Generally, all the tested methods exhibited a great improvement in their authentication performance when they were implemented on the trace face-feature vectors instead of the original input images. Two conclusions can be made from these results: a) the trace face-feature vector improves the performance of all the subspace projection techniques (due to the facts presented in Section 4), b) the proposed KPA technique can outperform GDA, KFA and CKFA. The reason that KPA performed better than the other tested subspace methods might be found in the use of the KPLS subspace, which emphasizes the class membership information of the face images even before Fisher's discriminant analysis is applied, making the resulting features contain more discriminative information than the features extracted with GDA, KFA or CKFA. Based on these facts, our future research will be focused on incorporating the trace transform and KPA into a feature-extraction approach that makes use of Gabor wavelet-based features, which exhibit high discriminatory power and are particularly suited for face authentication.

References

[1] C. Liu, ”Gabor-based kernel PCA with fractional power polynomial models for face recognition”, *TPAMI* **26** (5), 572–581 (2004).

[2] S.G. Kong, J. Heo, B.R. Abidi, J. Paik, M.A. Abidi, ”Recent advances in visual and infrared face recognition - a review”, *Computer Vision and Image Understanding* **97**, 103–135 (2005).

[3] K. Delac, M. Grgic, S. Grgic, ”Independent Comparative Study of PCA, ICA, and LDA on the FERET Data Set”, *International Journal of Imaging Systems and Technology* **15** (5), 252–260 (2006)

[4] M. Turk, A. Pentland, ”Eigenfaces for Recognition”, *Journal of Cognitive Neurosicence* **3** (1), 71–86 (1991).

[5] P.N. Belhumeur, J.P. Hespanha, D.J. Kriegman, ”Eigenfaces vs. fisherfaces: recognition using class specific linear projection”, *Proc. of the 4th European Conference on Computer Vision, ECCV'96*, Cambridge, UK, 45–58 (1996).

[6] J.W. Lu, K.N. Plataniotis, A.N. Venetsanpoulos, ”Regularized discriminant analysis for small sample size problem in face recognition”, *Pattern recognition letters* **24** (16), 3079–3087 (2003).

[7] C. Liu, H. Wechsler, ”Enhanced fisher linear discriminant models for face recognition”, *Proc. of the International Conference on Pattern Recogntion, ICPR'98*, 1368–1372 (1998).

[8] R. Lotlikar, R. Kothari, ”Fractional-step dimensionality reduction”, *TPAMI* **22** (6), 623–627 (2000).

[9] H. Yu, H. Yang, ”A direct LDA algorithm for high-dimensional data - with application to face recognition”, *Pattern recognition* **34** (10), 2067–2070 (2001).

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

[11] G. Baudat, F.E. Anouar, ”Generalized discriminant analysis using a kernel approach”, *Neural Computation* **12** (10), 2385–2404 (2000).

[12] Liu, C., ”Capitalize on dimensionality increasing techniques for improving face recognition grand challenge performance”, *TPAMI* **28** (5), 725–737 (2006).

[13] J. Yang, A.F. Frangi, J.Y. Yang, D. Zhang, Z. Jin, ”KPCA plus LDA: A complete kernel fisher discriminant framework for feature extraction anf recognition”, *TPAMI* **27** (2), 230–244 (2005).

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

[15] S. Srisuk, M. Petrou, W. Kurutach and A. Kadyrov, ”Face authentication using the trace transform”, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’03)*, Madison, Wisconsin, USA, 305–312 (2003).

[16] A. Kadyrov, M. Petrou, ”The trace transform and ist applications”, *TPAMI* **23** (8), 811–828 (2001).

[17] L. Shen, L. Bai, ”A review of Gabor wavelets for face recognition”, *Pattern Analysis and Applications* **9**, 273-292 (2006).

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

[19] L. Wiskott, J.M. Fellous, N. Kruger, C. von der Malsburg, ”Face recognition by elastic bunch graph matching”, *TPAMI* **19**, 775–779 (1997).

[20] C. Liu, H. Wechsler, ”Gabor feature based calssification using the enhanced fisher linear discriminant model for face recognition”, *Transactions on Image Processing* **11** (4), 467–476 (2002).

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

[22] S. Srisuk, M. Petrou, W. Kurutach and A. Kadyrov, ”A face authentication system using the trace transform”, *Pattern Analysis and Applications* **8**, 50–61 (2005).

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

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

[25] R. Rosipal, ”Kernel partial least squares for nonlinear regression and discrimination”,*Neural network world* **13** (3), 291–300 (2003).

[26] J. Baek, M. Kim, ”Face recognition using partial least squares components”, *Pattern Recognition* **37**, 1303–1306 (2004).

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

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

[29] W. Xuan, X. Bin, M. Jian-feng, B. Xiu-li, ”Scaling and rotation invariant analysis approach to object recognition based on radon and fourier-mellin transforms”, *Pattern Recognition*, Accepted for publishing, (2007).

[30] J. Kittler, Y. Li, J. Matas, ”On Matching Scores for LDA-based Face Verification”, *British mashine vision conference, BMVC’00*, Bristol, UK, 42–51 (2000).

[31] K. Messer, J. Matas, J. Kittler, J. Luettin, G. Maitre, ”XM2VTSDB: the extended M2VTS database”, *Proc. of the 2nd international conference on audio and video-based biometric person authentication, AVBPA'99*, Washington D.C., 72–77 (1999).

[32] J. Luettin, G. Maitre, 1998. ”Evaluation protocol for the extended M2VTS database”, *Tech. Report IDIAP*.

[33] E. Rentzeperis, A. Stergiou, A. Pnevmatikakis, L. Polymenakos ”Impact of Face Registration Errors on Recognition”, *Artificial Intelligence Applications and Innovations*, Peania, Greece, (2006).

[34] R. Gross, S. Baker, I. Matthews, T. Kanade, ”Face Recognition Across Pose and Illumination”, In: *Handbook of Face Recognition*, S. Z. Li, Anil K. Jain, Eds., Springer-Verlag, (2004).

[35] A. Pozne, ”Extracting visual features for automatic speech recognition”, *PhD thesis*, Faculty of electrical engineering, University of Ljubljana, (2005).