Eigenspace-based Face Recognition: A comparative study of different hybrid approaches

Pablo NAVARRETE and Javier RUIZ-DEL-SOLAR
Department of Electrical Engineering, Universidad de Chile.
Email: {pnavarre, jruizd}@cec.uchile.cl

Abstract. Different eigenspace-based approaches have been proposed for the recognition of faces. They differ mostly in the kind of projection method been used, in the projection algorithm been employed, in the use of simple or differential images before/after projection, and in the similarity matching criterion or classification method employed. Statistical, neural, fuzzy and evolutionary algorithms are used in the implementation of those systems. The aim of this paper is to present an independent, comparative study between some of these hybrid eigenspace-based approaches. This study considers theoretical aspects as well as simulations performed using a small face database (Yale Face Database) and a large face database (FERET).

1. Introduction

Face Recognition is a high dimensional pattern recognition problem. Even low-resolution face images generate huge dimensional feature spaces (20,000 dimensions in the case of a 100x200 pixels face image). In addition to the problems of large computational complexity and memory storage, this high dimensionality makes very difficult to obtain statistical models of the input space using well-defined parametric models. Moreover, this last aspect is further stressed given the fact that only few samples for each class (1-4) are normally available for training the system. Hybrid eigenspace-based approaches have been used in the literature to overcome the mentioned problems. Given that similar troubles are normally found in many biometric applications, we believe that some of the hybrid eigenspace-based methods to be outlined and compared in this work can be applied in the implementation of other biometric systems (signature, fingerprint, iris, etc.).

Among the most successful approaches used in face recognition we can mention eigenspace-based methods, which are mostly derived from the Eigenface-algorithm [14]. These methods project input faces onto a dimensional reduced space where the recognition is carried out, performing a holistic analysis of the faces. Different eigenspace-based methods have been proposed. They differ mostly in the kind of projection/decomposition approach been used (standard-, differential- or kernel-eigenspace), in the projection algorithm been employed, in the use of simple or differential images before/after projection, and in the similarity matching criterion or classification method employed. The aim of this paper is to present an independent, comparative study among some of these different approaches. We believe that to carry out an independent study is important, because comparisons are normally performed using the own implementations of the research groups that have proposed each method (e.g. in FERET contests), which does not consider completely equal working conditions (e.g. exactly the same pre-processing steps).

Very often, more than a comparison between the capabilities of the methods, a contest between the abilities of the research groups is performed. Additionally, not all the possible implementations are considered (e.g. p projection methods with q similarity criteria), but only the ones that some groups have decided to use.

This study considers standard, differential and kernel eigenspace methods. In the case of the standard ones, three different projection algorithms (Principal Component Analysis - PCA, Fisher Linear Discriminant - FLD and Evolutionary Pursuit - EP) and five different similarity matching criteria (Euclidean-, Cosines- and Mahalanobis-distance, SOM-Clustering and Fuzzy Feature Contrast - FFC) were considered. In the case of differential eigenspace methods, two approaches were used, the pre-differential [4] and the post-differential one [9]. In both cases two classification methods, Bayesian and Support Vector Machine – SVM classification, were employed. Finally, regarding kernel eigenspace methods [6], Kernel PCA - KPCA and Kernel Fisher Discriminant - KFD were used.

It is important to point out that in the mentioned approaches a hybrid utilization of statistical, neural, fuzzy and evolutionary algorithms is used. Thus PCA, FLD, KPCA, KFD and the Bayesian classifier correspond to statistical algorithms. SOM and SVM are two well-known neural models. And finally, FFC and EP correspond to fuzzy and evolutionary algorithms, respectively. In this context these algorithms are used in a complementary more than in a competitive way. Requirements as robustness, higher recognition rates, tolerance for imprecision and uncertainty, flexibility, user-friendship and low-cost can be fulfilled by the join use of them.

This comparative study considers theoretical aspects as well as simulations performed using the Yale Face Database, a database with few classes and several images per class, and FERET, a database with many classes and few images per class. It is important to use both kind of databases, because, as it will be shown in this work, some properties of the methods, as for example their generalization ability, changes depending on the number of classes taken under consideration.

The pre-processing aspects of all the approaches (face alignment, illumination invariance, geometrical invariance, etc.) are kept unchanged among their different implementations.

This paper is structured as follows. In section 2 several hybrid approaches for the eigenspace-based recognition of faces are described. In section 3 a comparative study among these hybrid approaches is presented. Finally, some conclusions of this work are given in section 4.

2. Hybrid Approaches for the eigenspace-based Recognition of Faces

Standard eigenspace-based approaches project input faces onto a dimensional reduced space where the recognition is carried out. In 1987 Sirovich and Kirby used PCA in order to obtain a reduced representation of face images [13]. Then, in 1991 Turk and Pentland used the PCA projections as the feature vectors to solve the problem of face recognition, using the Euclidean distance as the similarity function [14]. This system was the first eigenspace-based face recognition approach and, from then on, many eigenspace-based systems have been proposed using different projection methods and similarity functions. A differential eigenspace-based approach that allows the application of statistical analysis in the recognition process, was proposed in 1997 by Pentland and Moghaddam [4]. The main idea is to work with differences between face images, rather than with face images. In this way the recognition problem becomes a two-class problem, because the so-called “differential image” contains information of whether the two subtracted images are of the same class or different classes. In this case the number of training images per class increases so that statistical information becomes available. The system proposed in [4] used Dual-PCA projections and a Bayesian classifier. Following the same approach, a system using Single-PCA projections and a SVM classifier was proposed in [9].

In the differential case all the face images need to be stored in the database, which slow down the recognition process. This is a serious drawback in practical implementations. To overcome that drawback a so-called post-differential approach was proposed in [9]. Under this approach, differences between reduced face vectors are used instead of differences between face images. This allows decreasing the number of computations and the required storage capacity (only reduced face vectors are stored in the database), without losing the recognition performance of the differential approaches. Both, Bayesian and SVM classifiers were used to implement this approach in [9].

In the Kernel-based approaches KPCA and KFD, non-linear extensions of PCA and FLD respectively, are used as projection algorithms. The main idea behind this approach is to use linear methods applied to high-dimensional mapped vectors instead of the original vectors, and at the same time to avoid the explicit mapping of these vectors by means of the so-called "kernel-trick". The generalization of linear methods to non-linear ones using kernels, works as follows: if the algorithm to be generalized uses the training vectors only in the form of Euclidean dot-products, then it can be "kernelized", and all the dot-products like $\mathbf{x}^T\mathbf{y}$ are replaced by a so-called kernel function K($\mathbf{x}$,$\mathbf{y}$). If K($\mathbf{x}$,$\mathbf{y}$) fulfills the Mercer's condition, i.e. the operator K is semi-positive definite, then the kernel can be expanded into a series $\mathrm{K}(\mathbf{x},\mathbf{y}) = \sum_i \phi_i(\mathbf{x})\phi_i(\mathbf{y})$. In this way the kernel represents the Euclidean dot-product on a different space, called feature space F, on which the original vectors are mapped using the eigenfunctions $\boldsymbol{\phi}$: $\Re^N \rightarrow F$. Depending on the kernel function been used, the feature space F can be even of infinite dimension, as the case of Radial Basis Function (RBF) kernel, but we are never working in such space. A unified framework for solving kernel methods was proposed in [10], including the solution of multiclass-KFD, previously unknown. A unified Kernel-based face recognition system was proposed in [6] [7].

Standard-, differential- and kernel-eigenspace approaches for the recognition of faces are described in the following subsections.

2.1. Standard Eigenspace Face Recognition

Fig. 1 shows the block diagram of a generic, standard eigenspace-based face recognition system. Standard eigenspace-based approaches approximate the face vectors (face images) by lower dimensional feature vectors. These approaches consider an off-line phase or training, where the *projection matrix* ($\mathbf{W} \in R^{N\times m}$), the one that achieve the dimensional reduction, is obtained using all the database face images. In the off-line phase are also calculated the *mean face* ($\overline{\mathbf{x}}$) and the reduced representation of each database image (p^k). These representations are the ones to be used in the recognition process. The projection methods employed in this work for the reduction of dimensionality are PCA [14], FLD [2] and EP [3]. The similarity matching criteria used for the recognition process are Euclidean-, Cosine- and Mahalanobis-distance, SOM-Clustering, and FFC similarity. All these methods have been analyzed in [8]. Under this standard eigenspace approach a *Rejection System* for unknown faces is implemented by placing a threshold over the similarity measure.

2.2 Differential Eigenspace Face Recognition

Fig. 2 shows the block diagram of a generic, pre-differential eigenspace-based face recognition system. In this approach the whole face images are stored in the database (*NT* images). Previously the database face images are centered and scaled so that they are aligned. An input face image is preprocessed and subtracted from each database image. The result of each subtraction is called "differential image" Δ in R^N and it is the key for identification. That because it contains information of whether the two subtracted images are of the same class or different classes. In this way the original problem of *NC* classes becomes a two-class problem. The so-called differential images (*NT*) are projected into a reduced space using a given projection method. Thus, each image is transformed into a reduced differential vector δ in R^m. Thereafter the classification of the reduced differential vectors is performed. The result of each classification (S_i) is negative if the subtracted images (each δ) are of different classes and positive in other case. In order to determine the class of the input face image, the reduced vector with maximum positive classification value is chosen, and the class of its initial database image is given as the result of identification. The rejection system acts just when the maximum classification value is negative, i.e. it corresponds to the subtraction of different classes. Dual-PCA and Single-PCA projections have been used as projection methods. The Dual-PCA projections employ two projection matrices: $\mathbf{W}_I \in R^{N\times m_I}$ for intra-classes Ω_I (subtractions within equal classes), and $\mathbf{W}_E \in R^{N\times m_E}$ for extra-classes Ω_E (subtractions between different classes). Dual-PCA projections are employed together with a Bayesian classifier in order to perform the classification of the differential images [4]. Single-PCA projection employs a single projection matrix $\mathbf{W} \in R^{N\times m}$ (standard PCA) that reduces the dimension of the differential face images, and it is used together with a SVM classifier in order to perform the classification [9].

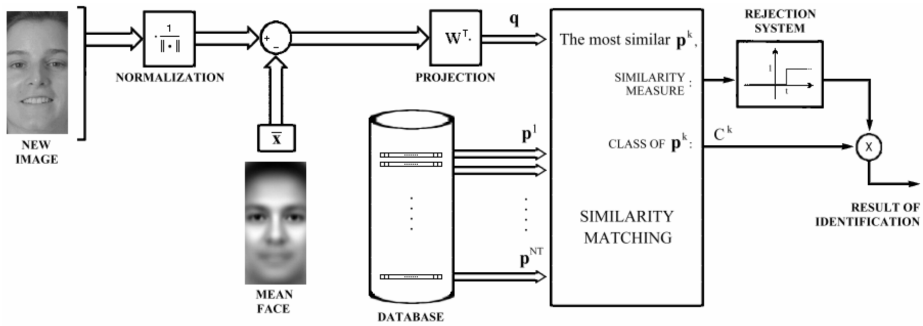

Fig. 1. Block diagram of a generic, standard eigenspace-based face recognition system.

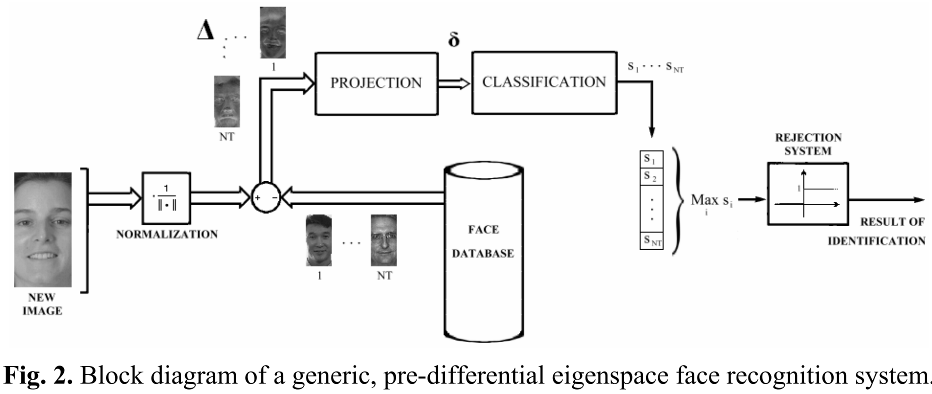

Fig. 2. Block diagram of a generic, pre-differential eigenspace face recognition system.

Fig. 3 shows the block diagram of a generic, post-differential eigenspace-based face recognition system. In this approach only the reduced face images are stored in the database (*NT*). An input face image is preprocessed and then projected into a reduced space using a given projection method. Thereafter, the new reduced face image is subtracted from each database reduced face image. The result of each subtraction is called "post-differential image" δ in R^m. This vector contains information of whether the two subtracted vectors are of the same class or different classes (intra-classes or extra-classes), and then it works in the same way as a "differential image" once projected on the reduced space. The classification module performs the classification of the post-differential vectors (*NT*). The class of the reduced database vector that has the maximum positive classification value gives the class of the initial input face image. If the projection module does not significantly change the topology of the differential-image space, then the pre-differential and post-differential approaches should have very similar recognition rates. The rejection system acts just when the maximum classification value is negative, i.e. it corresponds to the subtraction of different classes. In [9] two different systems that follow this approach were implemented. The first uses Single-PCA projections together with a Bayesian classifier, while the second employs Single-PCA projections together with a SVM classifier.

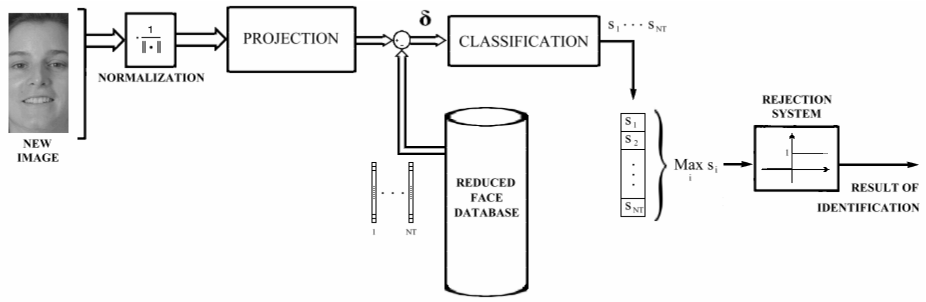

Fig. 3. Block diagram of a generic, post-differential eigenspace face recognition system.

2.3 Kernel Eigenspace Face Recognition

Under this approach the projection of the input face images is carried out in two steps [6]. In the first step, each preprocessed input face image $\mathbf{x} \in \Re^N$ (mapped onto the feature space $\phi(\mathbf{x}) \in F$) is projected on the *NT* support images (mapped in the same feature space), using the kernel function $K : \Re^N \times \Re^N \to \Re^{NT}$. For doing that in the face database the *NT* face images needs to be stored. This is due to the fact that kernel machines need all the image vectors in order to reproduce the eigenvectors in the feature space *F* [10]. In the second step, the parameters of the given kernel machine $\mathbf{A}^T \in M^{NT\times m}$ (see details in [10]), are applied to the kernel projection vector $\mathbf{k} \in \Re^{NT}$, in order to obtain the feature vector $\mathbf{q} \in \Re^m$. Afterwards, the *Similarity Matching* module compares the similarity of the reduced representation of the query face vector $\mathbf{q}$ with the reduced vectors $\mathbf{p}^k$, $\mathbf{p}^k \in \Re^m$, that correspond to reduced vectors of the faces in the database. Therefore in the database the reduced vectors needs also to be stored. By using a given criterion of similarity the *Similarity Matching* module determines the most similar vector $\mathbf{p}^k$ in the database. The class of this vector is the result of the recognition process, i.e. the identity of the face. In addition, a *Rejection System* for unknown faces is used if the similarity matching measure is not good enough. The exact computation of *K*() and $\mathbf{A}^T$ is described in [6][10].

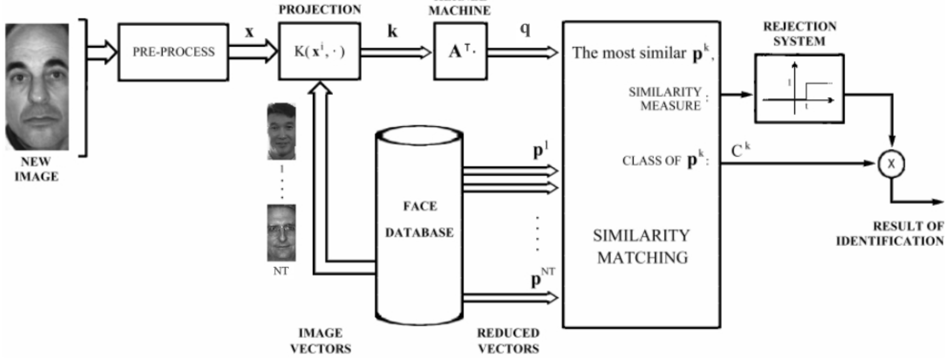

Fig. 4. Block diagram of a generic, kernel-based face recognition system.

3. *Comparative Study*

The comparative study presented in this section considers theoretical aspects as well as simulations performed using the Yale Face Database, a database with few classes and several images per class, and FERET, a database with many classes and few images per class. It is important to use both kinds of databases, because some properties of the methods, as for example their generalization ability, change depending on the number of classes under consideration. The pre-processing aspects of all the approaches (face alignment, illumination invariance, geometrical invariance, etc.) are kept unchanged among their different implementations.

3.1. Pre-processing

The preprocessing includes: window resizing (scale the face image using fixed proportions, to obtain face images of 100×200), masking (to avoid border pixels not corresponding to the face in the image), illumination gradient compensation (subtract a best-fit brightness plane to compensate heavy shadows caused by extreme lighting angles), histogram equalization (to spread the energy of all intensity values on the image), and normalization (to make all input images of the same energy). All these operations are detailed described in [7].

3.2 Simulations using the Yale Face Image Database

The first simulations were carried out using the Yale University - Face Image Database [15]. We employed 150 images of 15 different classes. First, we pre-processed the images manually by masking them in windows of 100x200 pixels and centering the eyes in the same relative places. In table 1 we show the results of several simulations for the standard eigenspace approaches. For each simulation we used a fixed number of training images, using the same type of images per class, according with the Yale database specification. In order to obtain representative results we take the average of 20 different sets of images for each fixed number of training images. All the images not used for training were used for testing. In tables 2 and 3 we show the results of several simulations using pre-differential and post-differential approaches. We used equal *a priori* probabilities for the Bayes-based methods, $P(\Omega_I) = P(\Omega_E)$, and a penalty for non-separable cases $C = 0.01$ in the SVM classification method (see details in [9]). In table 4 we show results using kernel methods. The kernel function was a RBF with σ=0.5. In KFD the regularization parameter was μ=0.05 (see details in [10]). In tables 1-4 the best results obtained in each experiment, for each approach (standard, pre- and post-differential and kernel), are indicated in bold.

3.3 Simulations using FERET

In order to test the described approach using a large database, we made simulations using the FERET database [12]. We use a target set with 762 images of 254 different classes (3 images per class), and a query set of 254 images (1 image per class). Eyes' location is included in the FERET database for all the images being used. Then the pre-processing considers centering and scaling images so that eyes' position keeps in the same relative place. In table 5 we show the results of simulations for the standard approaches. In this table the SOM-based clustering was not included because in these tests the number of classes (254) is much larger than the number of images per class (3), and the training process is very difficult. In tables 6 and 7 we show the results of simulations using pre-differential and post-differential approaches. For the Bayesian and SVM classifiers were used the same parameters as before (see 3.2). In table 8 we show results using kernel methods. The kernel function was a RBF with σ=0.5. In KFD the regularization parameter was μ=0.05 (see details in [10]). In tables 5-8 the best results obtained in each experiment, for each approach (standard, pre- and post-differential and kernel), are indicated in bold.

Tables 1, 2, 3 and 4. Mean recognition rates using the Yale database and different numbers of training images per class, and taking the average of 20 different training sets. The small numbers are standard deviations. All results consider the top 1 match. Whitening is equivalent to use a Mahalanobis distance in a projection space [8].

Table 1. Standard Eigenspace.

projection method	images per class	axes	Euclidean	cos()	SOM	FFC	whitening Euclidean	whitening cos()	whitening SOM	whitening FFC
PCA	6	49	95,7 2,7	95,8 2,7	94,2 2,8	81,8 5,4	83,3 5,9	89,3 4,1	88,8 3,8	81,8 5,4
FLD		14	94,6 2,1	95,2 2,5	93,5 2,4	85,9 5,2	**97,2** **2,2**	97,0 2,5	96,7 3,5	90,8 5,5
EP		21	89,8 4,1	94,3 4,0	92,7 4,2	85,2 3,8	-	-	-	-
PCA	5	42	94,0 2,5	94,1 2,5	92,5 3,1	76,8 10,4	82,2 7,2	87,7 5,6	87,3 5,8	76,8 10,4
FLD		14	94,0 3,2	94,3 2,6	92,8 2,8	87,3 5,8	**95,0** **3,9**	94,2 4,7	93,7 4,5	87,7 6,1
EP		16	93,7 3,2	93,9 2,6	92,1 3,3	88,2 3,6	-	-	-	-
PCA	4	35	93,4 1,9	93,4 2,1	91,6 2,3	78,7 5,5	85,4 4,0	88,0 4,0	79,2 5,3	78,7 5,5
FLD		14	92,9 2,4	93,5 2,4	93,8 2,5	84,7 3,8	**94,4** **2,1**	92,9 3,9	93,1 4,2	85,1 5,7
EP		14	92,3 2,6	92,9 2,5	91,8 2,6	85,3	-	-	-	-
PCA	3	28	91,9 2,5	92,4 2,2	88,5 2,7	78,6 6,8	84,2 4,0	86,0 4,5	84,0 5,6	78,6 6,8
FLD		14	89,8 4,5	90,9 4,4	85,7 4,6	81,6 5,5	**93,0** **2,2**	92,0 2,8	92,1 2,8	83,9 5,0
EP		14	84,2 3,5	91,2 4,4	88,5 4,4	82,1 5,1	-	-	-	-
PCA	2	21	88,9 5,0	88,9 5,0	79,1 6,1	75,5 6,9	83,4 6,3	85,1 4,0	75,2 6,6	75,5 6,9
FLD		14	88,5 3,4	88,1 4,2	77,2 4,2	79,6 6,1	**89,9** **4,1**	88,1 2,8	86,6 3,4	77,2 7,2
EP		14	77,9 4,8	88,4 5,1	78,5 3,9	78,4 5,3	-	-	-	-

Table 2. Pre-differential Eigenspace. (i)/(e) indicates intra/extra-classes.

images per class	Axes Dual-PCA	Axes SVM	Dual-PCA	SVM
6	168 (i) / 182 (e)	334	**97,5** **3,6**	96,8 3,4
5	115 (i) / 126 (e)	153	**95,5** **3,7**	94,6 4,1
4	85 (i) / 84 (e)	121	**93,2** **3,0**	92,1 3,4
3	39 (i) / 41 (e)	74	90,4 3,2	**91,8** **4,4**
2	12 (i) / 14 (e)	23	89,5 3,7	**90,3** **4,6**

Table 3. Post-differential Eigenspace.

images per class	axes	Dual-PCA	SVM
6	342	95,6 **4,3**	**97,4** 3,6
5	151	92,1 **5,3**	**93,8** 4,5
4	115	92,3 **4,2**	**92,6** 3,0
3	81	87,9 3,7	**91,5** **3,8**
2	22	86,7 4,8	**88,9** **5,1**

Table 4. Kernel Methods.

projection method	images per class	axes	Euclidean	cos()	SOM	FFC	whitening Euclidean	whitening cos()	whitening SOM	whitening FFC
KPCA	6	89	96,1	96,1	95,1	82,7	92,6	90,7	88,3	82,7
			2,7	2,7	2,6	8,9	4,6	5,9	6,3	8,9
KFD		14	**96,9**	96,8	95,1	92,4	96,3	93,9	94,5	89,8
			2,2	1,9	1,9	4,2	2,6	3,6	4,2	6,0
KPCA	5	74	94,5	94,5	92,3	82,9	88,9	87,7	87,9	82,9
			2,5	2,5	2,6	9,9	7,5	7,6	9,2	9,9
KFD		14	94,9	**95,4**	91,4	89,4	94,5	92,3	91,8	87,6
			4,1	**2,8**	2,9	5,2	4,1	5,4	5,6	6,1
KPCA	4	59	93,7	93,7	90,6	84,6	89,9	88,1	83,4	84,6
			1,9	1,9	2,2	6,2	4,9	4,6	5,8	6,2
KFD		14	94,1	**95,7**	91,5	89,1	92,9	91,5	91,5	84,3
			3,4	**2,4**	3,4	4,3	2,6	3,7	4,1	4,9
KPCA	3	44	92,5	92,5	90,5	82,6	90,3	88,1	83,8	82,6
			1,9	1,9	2,5	5,5	3,3	3,6	5,4	5,5
KFD		14	92,7	**94,0**	90,3	87,1	93,0	91,3	90,2	82,3
			2,6	**1,8**	2,0	5,2	2,4	3,0	3,3	4,8
KPCA	2	29	89,9	89,9	85,2	76,2	90,2	87,7	83,3	76,2
			4,3	4,3	4,4	7,8	3,7	3,2	7,5	7,8
KFD		14	90,4	**92,3**	88,4	82,0	89,1	87,3	84,5	77,5
			2,6	**3,6**	3,6	4,7	4,1	4,1	5,4	5,7

Tables 5, 6, 7 and 8. Mean recognition rates for standard approaches using FERET. All results consider the top 1 match for recognition. Whitening is equivalent to use a Mahalanobis distance in a projection space [8].
6

Table 5. Standard Eigenspace.

projection method	images per class	axes	Euclidean	cos()	FFC	whitening Euclidean	whitening cos()	whitening FFC
PCA	3	316	**94,1**	**94,1**	87,4	77,6	92,5	87,4
FLD		253	92,5	92,1	91,7	79,9	92,9	90,9
EP		218	92,3	91,8	91,5	-	-	-
PCA	2	252	86,4	**86,8**	81,5	73,2	85,6	81,5
FLD		253	85,2	85,0	82,9	73,4	79,9	83,1
EP		194	85,7	86,3	83,7	-	-	-

Table 6. Pre-differential Eigenspace.
(i)/(e) indicates intra/extra-classes.

images per class	Axes Dual-PCA	Axes SVM	Dual-PCA	SVM
3	148 (i) / 156 (e)	247	94,2	**94,8**
2	106 (i) / 128 (e)	192	**89,0**	88,7

Table 7. Post-differential Eigenspace.

images per class	axes	Dual-PCA	SVM
3	253	92,1	**95,2**
2	199	87,3	**88,5**

Table 8. Kernel Methods.

projection method	images per class	axes	Euclidean	cos()	FFC	whitening Euclidean	whitening cos()	whitening FFC
KPCA	3	761	94,5	94,5	85,4	79,5	95,3	85,4
KFD		253	95,3	94,5	**95,7**	82,7	75,2	60,2
KPCA	2	507	86,6	86,6	83,3	**89,6**	89,4	83,3
KFD		253	87,8	87,8	88,6	77,2	71,9	62,0

3.4 Results Analysis

By analyzing the Yale-database simulations (tables 1-4), the following can be concluded:

- When using the standard approach, the best results are obtained with the FLD-Withening-Euclidean combination. Using other FLD combinations very similar results are obtained (consider the standard deviation information). FLD uses also less projection axes than the other algorithms.
- Considering the differential approaches, the results obtained using the pre-differential approach are slightly better than the ones obtained with the post-differential one. When the number of images per classes is low (2 or 3) the results for both approaches are very similar (consider the standard deviation information). Dual-PCA and SVM gives similar results in both cases.
- KFD gives better results than KPCA using less projection axes. Very similar results are obtained using Euclidian or Cosines distances.
- The Yale database contains few classes and several images per class (2-6), and in general the best results obtained with each approach are very similar. Differences are seen only when the number of images per classes is low (2 or 3). In this last case, the approaches with better generalization ability, that is the kernel ones, obtains better results. By looking at the standard deviation information it can also be noted that the kernel approaches have a smaller variability in their results. Regarding the number of projection axes employed, the standard approaches use less axes.

By analyzing the FERET-database simulations (tables 5-8), the following can be concluded:

- When using the standard approach, the best results are obtained with PCA. As similarity measure Euclidian or Cosines distances can be used. The number of projection axes used in each approach is of the same order.
- Considering the differential approaches, the results obtained using the pre- and the post-differential approaches are almost identical. In both cases Dual-PCA and SVM gives similar results.
- Results obtained with KFD and KPCA are very similar. However, KFD gives better results when the number of images per class is 3. On the other hand, when the number of images per class is 2, the best results are obtained with KPCA. KFD uses less projection axes than KPCA.
- The FERET database contains many classes and few images per class (2-3), for this reason the best results are obtained when using the approaches with better generalization capabilities, i.e. the kernel ones.

Other information that should also be considered:

- Post-differential approaches are 2 to 5 times faster than the pre-differential ones. Taking in to account their similar recognition rates, post-differential approaches are the better differential alternative.
- Kernel-projections are 2 to 3 times slower than linear projections due to the use of the support images (all the database images). Another drawback of these methods is that the kernel parameters adjustment is very difficult and data dependant.

4. Conclusions

The aim of this paper was to present an independent, comparative study among different eigenspace-based approaches. The study considered standard, differential and kernel eigenspace methods. In the case of the standard ones, three different projection algorithms (PCA, FLD and EP) and five different similarity matching criteria (Euclidean-, Cosines- and Mahalanobis-distance, SOM-Clustering and FFC-similarity) were considered. In the case of the differential methods, two approaches were used, the pre-differential and the post-differential. In both cases Bayesian and SVM classification were employed. Finally, regarding kernel methods, KPCA and KFD were used.

Simulations were performed using the Yale Face Database, a database with few classes and several images per class, and FERET, a database with many classes and few images per class. By looking at the obtained results is can be concluded:

- Considering recognition rates, generalization ability and processing time, the best results were obtained with the post-differential approach, using either Dual-PCA or SVM.
- In the specific case of the Yale Face Database, where the requirements are not so high, using any of the approaches gives similar results.
- The drawbacks of the kernel methods are their low processing speed and the difficulty to adjust the kernel parameters. The first drawback could be overcome by using a kind of support vectors in the KPCA and KFD algorithms. For sure this is a problem to be tackled by kernel-machine researchers in the next few years.

As future work we will like to extend our study by considering other kernel approaches and algorithms, as for example Independent Component Analysis – ICA and Kernel-ICA.

Acknowledgements

This research was funded by the join "Program of Scientific Cooperation" of CONICYT (Chile) and BMBF (Germany) under the project "Face Recognition and Signature Verification using Soft-Computing" and by Millenium Nucleous Center for Web Research, Grant P01-029-F, Mideplan, Chile. Portions of the research in this paper use the FERET database of facial images collected under the FERET program.

References

[1] C.J.C. Burges, "A tutorial on support vector machines for pattern recognition", *Data Mining and Knowledge Discovery*, 2(2), pp. 121–167, 1998.
[2] R.A. Fisher, "The Use of Multiple Measures in Taxonomic Problems", *Ann. Eugenics*, vol. 7, pp. 179-188, 1936.
[3] C. Liu and H. Wechsler , "Evolutionary Pursuit and Its Application to Face Recognition", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 6, pp. 570-582, June 2000.
[4] A. Pentland and B. Moghaddam , "Probabilistic Visual Learning for Object Representation", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 696-710, July 1997.
[5] R.O. Duda, P.E. Hart and D.G. Stork, "Pattern Classification", Second Edition, 2001.
[6] P. Navarrete and J. Ruiz-del-Solar, "Kernel-based Face Recognition by a Reformulation of Kernel Machines", *Proc. of the 7th Online World Conf. on Soft Computing in Industrial Applications*, WSC 7, September 2002.
[7] P. Navarrete, "Face Recognition and Face Retrieval using Statistical Learning Techniques", Thesis for the title of Electrical Engineer, Department of Electrical Engineering, Universidad de Chile, June 2002 (in Spanish).
[8] P. Navarrete and J. Ruiz-del-Solar, "Comparative study between different Eigenspace-based approaches for Face Recognition", *Lecture Notes in Artificial Intelligence 2275*, AFSS 2002, Springer, 178 - 184.
[9] J. Ruiz-del-Solar and P. Navarrete, "Toward a Generalized Eigenspace-based Face Recognition System", *Lecture Notes in Computer Science* 2396 (Structural, Syntactic, and Statistical Pattern Recognition), Springer, 662 – 671.
[10] P. Navarrete and J. Ruiz-del-Solar, "On the Generalization of Kernel Machines", *Lecture Notes in Computer Science* 2388 (Pattern Recognition with Support Vector Machines - SVM 2002), Springer, 24 - 39.
[11] C. Cortes and V. Vapnik, "Support Vector Networks", *Machine Learning*, 20, pp. 273-297, 1995.
[12] P. J. Phillips, H. Wechsler, J. Huang and P. Rauss, "The FERET database and evaluation procedure for face recognition algorithms", *Image and Vision Computing J.*, Vol. 16, no. 5, 295-306, 1998.
[13] L. Sirovich and M. Kirby, "A low-dimensional procedure for the characterization of human faces", *J. Opt. Soc. Amer. A*, vol. 4, no. 3, pp. 519-524, 1987.
[14] M. Turk and A. Pentland, "Eigenfaces for Recognition", *J. Cognitive Neuroscience*, vol. 3, no. 1, pp. 71-86, 1991.
[15] Yale University Face Image Database, publicly available for non-commercial use, http://cvc.yale.edu/projects/yalefaces/yalefaces.html.