Pruning Noisy Bases in Discriminant Analysis*

Manli Zhu and Aleix M. Martínez
Depart. of Electrical and Computer Engineering
The Ohio State University, Columbus, OH 43210

Abstract

The success of Linear Discriminant Analysis (LDA) is due in part to the simplicity of its formulation, which reduces to a simultaneous diagonalization of two symmetric matrices $\mathbf{A}$ *and* $\mathbf{B}$*. However, a fundamental drawback of this approach is that it cannot be efficiently applied wherever the matrix* $\mathbf{A}$ *is singular or when some of the smallest variances in* $\mathbf{A}$ *are due to noise. In this paper, we present a factorization of* $\mathbf{A}^{-1}\mathbf{B}$ *and a correlation-based criterion that can be readily employed to solve these problems. We provide detailed derivations for the linear and non-linear classification problems. The usefulness of the proposed approach is demonstrated thoroughly using a large variety of databases.*

Keywords: Linear discriminant analysis, kernel discriminant analysis, data noise, discriminant power, principal components analysis, pattern recognition.

1 Introduction

Simultaneous diagonalization of two matrices is a powerful tool applicable to a large variety of problems in computer vision, bioinformatics and pattern recognition. This is usually achieved by means of an eigenvalue decomposition of $\mathbf{A}^{-1}\mathbf{B}$, where $\mathbf{A}$ and $\mathbf{B}$ are two symmetric matrices and $\mathbf{A}$ is positive-semidefinite [9, 17]. Discriminant analysis algorithms employ this to find those bases that minimize the metric defined by $\mathbf{A}$ and maximize the metric given by $\mathbf{B}$. Arguably, the most known algorithm using this basic approach is Fisher's Linear Discriminant Analysis (LDA) [7, 18], which assumes all classes can be described using a single Gaussian distribution with common covariance matrix. But alternatives to this formulation exist [9, 10, 22, 14, 13, 17, 24, 5, 11]. All these approaches employ, nonetheless, the same eigenvalue decomposition,

$$\mathbf{A}^{-1}\mathbf{B}\mathbf{V} = \mathbf{V}\Lambda, \tag{1}$$

where $\mathbf{V}$ is a matrix whose columns describe the discriminant feature vectors and $\Lambda = diag(\lambda_1, \ldots, \lambda_d)$ is the diagonal matrix of associate discriminant variances, with $\lambda_1 \geq \cdots \geq \lambda_d$. Note that whenever $d > rank(\mathbf{B})$, the last $d - p_{\mathbf{b}}$ variances will be zero, where $p_{\mathbf{b}} = rank(\mathbf{B})$.

Some of the most known metrics for this approach are: the between-class scatter matrix as $\mathbf{B}$, and the sample covariance matrix as $\mathbf{A}$. These are given by $\mathbf{B} = \sum_{i=1}^{C} \frac{n_i}{n} (\mu_i - \mu)(\mu_i - \mu)^T$, and $\mathbf{A} = \frac{1}{n}\sum_{i=1}^{n} (\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T$, where $\mathbf{x}_i$ are the n sample feature vectors, μ_i is the mean of the

*IEEE Transactions on Neural Networks, 2008.

samples in class i, μ is the global mean, and n_i and C are the number of samples in class i and the number of classes, respectively. Although many alternatives to these exist [9, 24], they generally follow the property that $rank(\mathbf{B}) \leq rank(\mathbf{A})$, which will be assumed herein.

Although (1) requires that we compute the inverse of $\mathbf{A}$, in [17], we have shown that this is actually not necessary because

$$\mathbf{A}^{-1}\mathbf{B} = \sum_{i=1}^{p_\mathbf{b}} \sum_{j=1}^{p_\mathbf{a}} \frac{\lambda_{\mathbf{b}_i}}{\lambda_{\mathbf{a}_j}} (\mathbf{a}_j^T \mathbf{b}_i) \mathbf{a}_j \mathbf{b}_i^T, \tag{2}$$

where $p_\mathbf{a} = rank(\mathbf{A})$, $p_\mathbf{b} \leq p_\mathbf{a}$, and $\mathbf{a}_j$ and $\mathbf{b}_i$ are the eigenvectors of $\mathbf{A}$ and $\mathbf{B}$ with associated eigenvalues $\lambda_{\mathbf{a}_j}$ and $\lambda_{\mathbf{b}_i}$. This result will prove fundamental for us in the design of new algorithms, because it simplifies all derivations to the computation of the eigenvector decomposition of our two matrices $\mathbf{A}$ and $\mathbf{B}$. This result comes from the realization that $\sum_{j=1}^{p_\mathbf{a}} \mathbf{a}_j^T \mathbf{b}_i \mathbf{a}_j$ is the reconstruction of the vectors of the metric $\mathbf{B}$ to be maximized in the subspace defined by the metric $\mathbf{A}$. Therefore, the eigenvectors of $\mathbf{A}$ determine the possible solution space where $\mathbf{B}$ can search for a common solution with $\mathbf{A}$. This solution is given by the relevance of the eigenvalues of the two matrices, $\lambda_{\mathbf{b}_i}$ and $\lambda_{\mathbf{a}_j}$.

It is clear from (2) that the resulting discriminant feature vectors $\mathbf{V}$ will be dominated by the eigenvectors of $\mathbf{A}$ associated to small eigenvalues. This works well when the eigenvectors associated to small variances describe the dimensions where the discriminant information is. Unfortunately, many of these small variances correspond to noisy features. Here, by *noisy features* we mean those dimensions that do not contribute to discrimination. This can either be caused by noise in the data or by features uncorrelated with the classification parameter (e.g., illumination variations in object recognition). *Discerning which bases are due to noise and which to the true discrimination of the data – that is to say, the ones to prune and the ones to keep – is a fundamental problem in discriminant analysis.*

In attempting to solve this problem, researchers have previously looked at a related issue – that given by the singularity of $\mathbf{A}$. That is, when $\mathbf{A}$ cannot be inverted, Eq. (1) can be preceded by a Principle Component Analysis (PCA) step where the original dimensionality of the data is reduced to one of k-dimensions, i.e., $\mathbb{R}^d \longrightarrow \mathbb{R}^k$. Here, k is chosen to guarantee that the projection of $\mathbf{A}$ onto $\mathbb{R}^k$ is no longer singular, i.e., $d \geq rank(\mathbf{A}) \geq k$ [2, 20, 6, 9].

A well-known solution for selecting a value for k examines the plot of the eigenvalues (conveniently ordered from largest to smallest) and then searches for the "elbow" where the values fall sharply. However, there exist too many applications where the eigenvalue plot drifts without any obvious cutting point. When this is the case, the first k PCs that account for

$$r = 100 \frac{\sum_{j=1}^{k} \lambda_{\mathbf{a}_j}}{\sum_{j=1}^{p_\mathbf{a}} \lambda_{\mathbf{a}_j}}$$

per cent of the total variance can be used. Although, r must be specified by the user, in [10] it is suggested that a value of r between 70% and 90% preserves most of the information needed for representing Gaussian-like densities. But determining the most convenient value is problem-specific.

An alternative to the method just described, is to define a criterion that accounts for the discriminant information on each basis vector $\mathbf{a}_j$. Most notably, [4, 10] define the discriminatory power of a vector $\mathbf{a}_j$ as

$$J(\mathbf{a}_j) = \frac{\mathbf{a}_j^T \mathbf{B} \mathbf{a}_j}{\lambda_{\mathbf{a}_j}}.$$

In this case, the numerator measures the agreement between the eigenvectors of $\mathbf{A}$ and those of $\mathbf{B}$, while the denominator is used to account for the variances. To see this note that $J(\mathbf{a}_j)$ can be rewritten

as

$$\sum_{i=1}^{p_{\mathbf{b}}} \frac{\lambda_{\mathbf{b}_i}}{\lambda_{\mathbf{a}_j}} \left(\mathbf{a}_j^T \mathbf{b}_i\right)^2, \tag{3}$$

which is the same as finding those $\mathbf{a}_j$'s that maximize the discriminant power in $\mathbb{R}^k$, because summing (3) over all $\mathbf{a}_j$ is the same as $tr\left(\mathbf{A}^{-1}\mathbf{B}\right)$ (see Theorem 2 in [17]).

The major problem with the solutions summarized in the preceding paragraphs, is that we do not know which of the eigenvectors of $\mathbf{A}$ are associated to noise, *because the eigenvalue does not distinguish between noisy features and bases describing the true discriminant information.* This means we need to define a criterion that is unbiased by the values of the variances associated to each of the bases of $\mathbf{A}$.

Close analysis of (3) reveals that the term $\left(\mathbf{a}_j^T \mathbf{b}_i\right)$ measures the correlation between the eigenvectors of $\mathbf{a}_j$ and the $ran(\mathbf{B})$; where $ran(\mathbf{M})$ is the range space of the matrix $\mathbf{M}$. To see this note that since the vectors $\mathbf{a}_j$ and $\mathbf{b}_i$ are unitary, their inner product defines the cosine of the angle, which is the correlation between them. This is in fact used by LDA to determine which basis vectors of $\mathbf{A}$ are correlated to the $ran(\mathbf{B})$. Only those that are correlated can contribute to the selection of discriminant bases $\mathbf{V}$. This product is hence used as a switch, to determine which bases of $\mathbf{A}$ contribute to the solution [17]. This is indeed the approach we will take in this paper. We will take advantage of this result to derive an approach for selecting those bases of $\mathbf{A}$ that are best suited for classification. Derivations for this method are in Sections 2 and 3. Experimental results are in Section 4. A preliminary version of this paper appeared in [23].

2 Pruning Noisy Bases

The spectral-decomposition of a symmetric matrix is $\mathbf{A} = \lambda_{\mathbf{a}_1}\mathbf{a}_1\mathbf{a}_1^T + \lambda_{\mathbf{a}_2}\mathbf{a}_2\mathbf{a}_2^T + \cdots + \lambda_{\mathbf{a}_{p_\mathbf{a}}}\mathbf{a}_{p_\mathbf{a}}\mathbf{a}_{p_\mathbf{a}}^T$, where $\lambda_{\mathbf{a}_1} \geq \lambda_{\mathbf{a}_2} \geq \cdots \geq \lambda_{\mathbf{a}_{p_\mathbf{a}}}$ are the eigenvalues of $\mathbf{A}$ and $\{\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_{p_\mathbf{a}}\}$ the corresponding eigenvectors. Such representations have been extensively used to carry out all sorts of data analysis. For example, when $\mathbf{A}$ is equal to the covariance matrix, one can select the first m eigenvectors to define the m-dimensional subspace that carries the largest amount of variance. Similarly, the eigenvectors associated to small variances can be used to detect near-constant linear relationships between variables [10].

Similar to the spectral decomposition of a matrix, we have shown that $\mathbf{A}^{-1}\mathbf{B}$ can be decomposed into the sum of eigenvectors and eigenvalues shown in (2). As above, this can now be used to eliminate some of the eigenvectors of $\mathbf{A}$ that are unnecessary or corrupted by noise. Following our notation, we define the reconstruction of $\mathbf{A}^{-1}\mathbf{B}$ in the subspace given by a set of k eigenvectors of $\mathbf{A}$ as

$$\left(\mathbf{A}^{-1}\mathbf{B}\right)_S = \sum_{\mathbf{a}_j \in S} \sum_{i=1}^{p_{\mathbf{b}}} \frac{\lambda_{\mathbf{b}_i}}{\lambda_{\mathbf{a}_j}} \mathbf{a}_j^T \mathbf{b}_i \mathbf{a}_j \mathbf{b}_i^T, \tag{4}$$

where S is a set of k elements drawn from $\{\mathbf{a}_1, \cdots, \mathbf{a}_{p_\mathbf{a}}\}$, without repetition. Further, there is a direct relation between the solution given by our above equation and that of the PCA-LDA algorithm (where PCA is used before applying LDA). This is shown in Appendix B.

Note that (4) defines a general form for the use of a subset of bases of $\mathbf{A}$. Depending on the criterion I used for such a selection, the final set of discriminant feature vectors $\mathbf{V}$ will vary; where now

$$\left(\mathbf{A}^{-1}\mathbf{B}\right)_S \mathbf{V} = \mathbf{V}\Lambda. \tag{5}$$

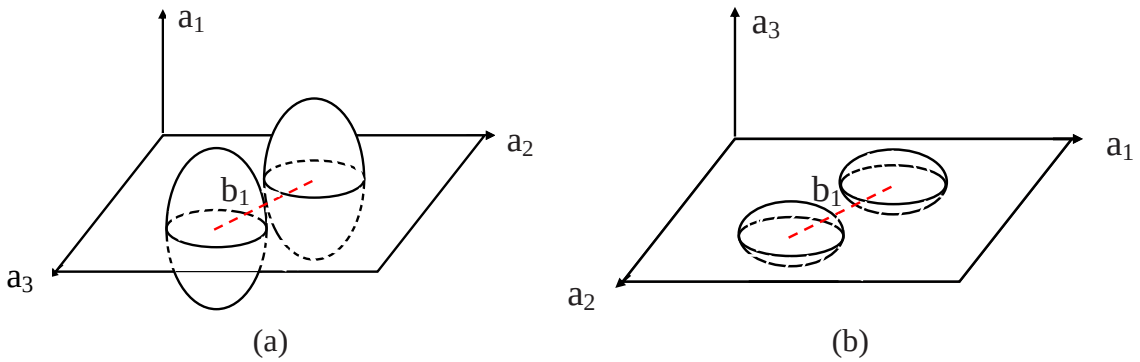

Figure 1: *In (a) the data has large variance about the* $\mathbf{z}$ *axis. In (b) this variance is small. In neither case* $\mathbf{z}$ *provides any discriminant information.*

The goal thus reduces to determining the criterion I that prunes those noisy bases of $\mathbf{A}$ but keeps those that contribute to the discrimination of the class distributions.

To see why (5) is equivalent to the pruning of noisy bases, one needs to realize that its discriminatory power is consistent with that of discriminant analysis. To see this, note that

$$tr\left(\mathbf{A}^{-1}\mathbf{B}\right)_S = \sum_{\mathbf{a}_j \in S} tr\left(\sum_{i=1}^{p_\mathbf{b}} \frac{\lambda_{\mathbf{b}_i}}{\lambda_{\mathbf{a}_j}} \mathbf{a}_j^T \mathbf{b}_i \mathbf{a}_j \mathbf{b}_i^T\right) = \sum_{\mathbf{a}_j \in S} \sum_{i=1}^{p_\mathbf{b}} \frac{\lambda_{\mathbf{b}_i}}{\lambda_{\mathbf{a}_j}} \left(\mathbf{a}_j^T \mathbf{b}_i\right)^2 = \sum_{\mathbf{a}_j \in S} J(\mathbf{a}_j).$$

The problem to be solved next is that of defining a criterion I that distinguishes between the noisy and discriminant bases. We do this in the section to follow. Before we do this though, it is appropriate to note that this method can be extended to other related applications, such as in canonical correlates analysis or principal components of instrumental variables. In these cases, we would only need to change the discriminant score, by its appropriate measurement.

3 Correlation-based Criterion

The decomposition shown in (2) will now prove instrumental for the understanding of the role the eigenvectors of each matrix play in discriminant analysis. In (2), $\mathbf{a}_j^T \mathbf{b}_i$ acts as a "switch," controlling which $\mathbf{a}_j$'s are correlated to the $\mathbf{b}_i$'s. To be able to play a role in the selection of discriminant vectors $\mathbf{V}$, $\mathbf{a}_j$ has to be correlated to the $ran(\mathbf{B})$, i.e., $\mathbf{a}_j^T \mathbf{b}_i \neq 0$ for at least one i in $\{1, \cdots, p_\mathbf{b}\}$, regardless of the value of $\lambda_{\mathbf{a}_j}$. This is illustrated in Fig. 1. In (a) the two distributions have a large variance about the $\mathbf{a}_1$ axis. In (b) this variance is close to zero. In both cases, however, the eigenvector aligned with $\mathbf{a}_1$ in (a) and with $\mathbf{a}_3$ in (b) is not correlated to the $ran(\mathbf{B})$ (which in this case is the outlined plane). This means that the eigenvector aligned with this dimension *can be eliminated from consecutive computations because this does not (and cannot) carry any discriminant information*. In the examples shown in Fig. 1, S will thus be constructed with those eigenvectors that define the $\mathbf{x}$-$\mathbf{y}$-plane only (where $\mathbf{x}$ and $\mathbf{y}$ are the bases $\mathbf{a}_2$ and $\mathbf{a}_3$ in the first example on the left and $\mathbf{a}_1$ and $\mathbf{a}_2$ in the example on the right).

The above result can be easily formulated in the form of a correlation-based measure as

$$I_j = \sum_{i=1}^{p_\mathbf{b}} \left(\mathbf{a}_j^T \mathbf{b}_i\right)^2, \quad 1 \leq j \leq p_\mathbf{a}. \tag{6}$$

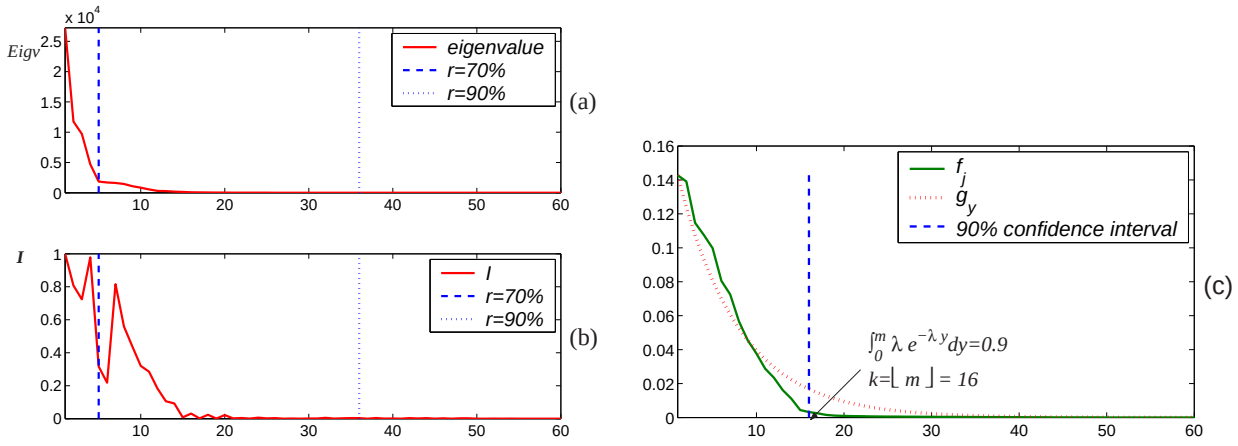

Figure 2: *(a) Plotted here are the eigenvalues of the sample covariance matrix of the ETH-80 database in decreasing order. (b) Shows the correlation values of each of the eigenvectors of* $\mathbf{A}$ *(ordered by eigenvalue) with those of* $\mathbf{B}$*. Dashed and dotted lines represent the cutoff points with* 70% *and* 90% *of the total variance. (c) The confidence interval of the exponential function* $g_y = \lambda e^{-\lambda y}$ *with probability* 0.9*, where* $\lambda = f_1 = \frac{I_1}{p_{\mathbf{b}}} = 0.1428$*.*

To illustrate the use of this measure, we will employ the data of the ETH-80 database [12]. The ETH-80 database contains images of eight object categories. Each category includes images of ten objects, each photographed from 41 distinct orientations. For each of these images, we obtain the two-dimensional silhouette of the object (i.e., the contour that separates the object from the background) and compute its centroid. Then, we calculate the length of the lines connecting this centroid with a set of equally separated points in the silhouette. In our experiments we obtain a total of 300 distances, generating a feature space of 300 dimensions. This allows us to calculate the between-class scatter matrix (to represent the metric given by $\mathbf{B}$) and the sample covariance matrix of the data (as $\mathbf{A}$). The eigenvectors of $\mathbf{A}$ can now be sorted in decreasing order of eigenvalue. This is shown in Fig. 2(a), where we only show the eigenvalue plot for the first 60 eigenvectors for clarity. In this plot, we also show dashed and dotted lines to represent the cut at 70% and 90% of the total variance.

Maintaining this order of eigenvectors, we calculate the correlation of each eigenvector $\mathbf{a}_j$ to $ran(\mathbf{B})$ using Eq. (6). These values are plotted in Fig. 2(b), where the lines defining the 70% and 90% of the total variance are still shown. Keeping the same ordering of eigenvectors in (a) and (b) allows us to see that some of the eigenvectors associated to small eigenvalues are very correlated to the $ran(\mathbf{B})$. Therefore, eliminating all the eigenvectors with associated small variance is not a good idea, because these should contribute to the outcome $\mathbf{V}$. Furthermore, it is important to note that the other eigenvectors $\mathbf{a}_j$ that should most worry us are those that are only slightly correlated to $ran(\mathbf{B})$ but have an associated small eigenvalue. While these eigenvectors should not play a role in defining $\mathbf{V}$, some will, because their variance is sufficiently small to have an effect in (4). In summary, we should keep the eigenvectors $\mathbf{a}_j$ that are most correlated to $ran(\mathbf{B})$ and prune the rest. Hence, it is only natural to sort the eigenvectors $\mathbf{a}_j$ from most to least correlated with $ran(\mathbf{B})$.

The last problem that needs to be addressed within our framework is that of defining a simple, efficient way of selecting the cutoff point which specifies the bases to be pruned. This we can do with the help of a monotonically decreasing density function that approximates the correlation values, because this can (in turn) be employed to specify a confidence interval. This can be easily done,

because the sum of the correlations over all eigenvectors of $\mathbf{A}$ is equal to $p_{\mathbf{b}}$. This means that

$$\frac{1}{p_{\mathbf{b}}}\sum_{j=1}^{p_{\mathbf{a}}} I_j = \frac{1}{p_{\mathbf{b}}}\sum_{j=1}^{p_{\mathbf{a}}}\sum_{i=1}^{p_{\mathbf{b}}} \left(\mathbf{a}_j^T\mathbf{b}_i\right)^2 = 1,$$

which facilitates the definition of the following normalized correlation value

$$f_j = \frac{1}{p_{\mathbf{b}}} I_j = \frac{1}{p_{\mathbf{b}}}\sum_{i=1}^{p_{\mathbf{b}}} (\mathbf{a}_j^T\mathbf{b}_i)^2. \tag{7}$$

When the number of classes is large (i.e., $p_{\mathbf{b}}$ is large), the curve defined by the set $\{f_1, \ldots, f_{p_{\mathbf{a}}}\}$ is less steeped. This is because $ran(\mathbf{B})$ is a subspace of high dimensionality. Taking this into account and noting that when $p_{\mathbf{b}}$ is small the curve goes to zero much faster (since $ran(\mathbf{B})$ represents a subspace of low dimensionality), allows us to approximate this curve with an exponential function of the form

$$g_y = \lambda e^{-\lambda y},$$

where $\lambda = f_1$. In Fig. 2(c) we show how g_y can approximate the shape of the original curve defined by $\{f_1, \ldots, f_{p_{\mathbf{a}}}\}$, which corresponds to the eigenvectors of $\mathbf{A}$ computed earlier using the ETH-80 database. As seen in the figure, this approximation can now be used to calculate the *confidence interval* h of g_y as

$$\int_0^m \lambda e^{-\lambda y} dy = h.$$

This result is important because it can be used to find the most adequate number k of eigenvectors of $\mathbf{B}$ as $-e^{-\lambda y}|_0^m = h$, which means

$$\begin{aligned} 1 - e^{-\lambda m} &= h \\ m &= -\ln(1-h)/\lambda \\ k &= min(\lfloor m \rfloor, p_{\mathbf{a}}). \end{aligned} \tag{8}$$

In our algorithm, we will always select the k providing a 90% confidence interval. An example was shown in Fig. 2(c) with $k = 16$.

4 Kernel Extension

The solution derived above is limited to the cases where the data is linearly separable. Fortunately, we can extend our result to the non-linear case by including a non-linear mapping function $\phi(.)$ in the process. This function will serve to map the non-linearly separable data $\mathbf{X}$ into a higher-dimensional space $\mathcal{F}$ where the data $\mathbf{X}^\Phi$ is linearly separable. In this new space $\mathcal{F}$, we will have

$$\mathbf{A}^{\Phi^{-1}}\mathbf{B}^\Phi = \sum_{i=1}^{p_{\mathbf{b}}}\sum_{j=1}^{p_{\mathbf{a}}} \frac{\lambda_{\mathbf{b}_i}^\phi}{\lambda_{\mathbf{a}_j}^\phi} (\mathbf{a}_j^{\phi^T}\mathbf{b}_i^\phi)\mathbf{a}_j^\phi \mathbf{b}_i^{\phi^T}. \tag{9}$$

The eigenvectors and eigenvalues of our two metrics above are given by the symmetric Schur decompositions $\mathbf{V}_a^{\Phi^T}\mathbf{A}^\Phi\mathbf{V}_a^\Phi = \Lambda_a^\Phi$ and $\mathbf{V}_b^{\Phi^T}\mathbf{B}^\Phi\mathbf{V}_b^\Phi = \Lambda_b^\Phi$, with $\mathbf{V}_a^\Phi = \left(\mathbf{a}_1^\phi, \ldots, \mathbf{a}_p^\phi\right)$ and $\mathbf{V}_b^\Phi =$

$\left(\mathbf{b}_1^\phi, \ldots, \mathbf{b}_p^\phi\right)$. These two eigenvector matrices must be in the span of $\Phi(\mathbf{X})$, where $\Phi(\mathbf{X}) = (\phi(\mathbf{x}_1), \ldots, \phi(\mathbf{x}_n))$, because the statistics computed by each metric do not include the null space of the data. This means we can write $\mathbf{V}_a^\Phi = \Phi(\mathbf{X})\Gamma_a$ and $\mathbf{V}_b^\Phi = \Phi(\mathbf{X})\Gamma_b$, with $\Gamma_a = \left(\gamma_{a_1}, \ldots, \gamma_{a_p}\right)$ and $\Gamma_b = \left(\gamma_{b_1}, \ldots, \gamma_{b_p}\right)$.

The new metrics to be employed in the kernel space, $\mathbf{M}_a$ and $\mathbf{M}_b$, are given by $\Gamma_a^T\Phi(\mathbf{X})^T\mathbf{A}^\Phi\Phi(\mathbf{X})\Gamma_a = \Gamma_a^T\mathbf{M}_a\Gamma_a = \Lambda_a^\Phi$ and $\Gamma_b^T\Phi(\mathbf{X})^T\mathbf{B}^\Phi\Phi(\mathbf{X})\Gamma_b = \Gamma_b^T\mathbf{M}_b\Gamma_b = \Lambda_b^\Phi$, which provide their eigenvectors and eigenvalues. Two commonly employed metrics (in the kernel approach) are $\mathbf{M}_a = \frac{1}{n}\mathbf{K}\mathbf{K}$ and $\mathbf{M}_b = \frac{1}{n}\mathbf{K}\mathbf{W}\mathbf{K}$, where $\mathbf{K} = \Phi(\mathbf{X})^T\Phi(\mathbf{X})$ is the gram matrix, and $\mathbf{W} = (\mathbf{W}_l), l = 1, \cdots, C$, $\mathbf{W}_l$ is a $n_l \times n_l$ matrix with all terms equal to $\frac{1}{n_l}$ [1]. When using such matrices though, Γ_a and Γ_b have to be normalized to $\bar{\gamma}_{a_j} = \frac{\gamma_{a_j}}{\sqrt{\gamma_{a_j}^T\mathbf{K}\gamma_{a_j}}}$, and $\bar{\gamma}_{b_i} = \frac{\gamma_{b_i}}{\sqrt{\gamma_{b_i}^T\mathbf{K}\gamma_{b_i}}}$, such that $\mathbf{V}_a^{\Phi T}\mathbf{V}_a^\Phi = \mathbf{I}$ and $\mathbf{V}_b^{\Phi T}\mathbf{V}_b^\Phi = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix. This guarantees that $\mathbf{V}_a^\Phi$ and $\mathbf{V}_b^\Phi$ provide the set of eigenvectors of $\mathbf{A}^\phi$ and $\mathbf{B}^\phi$ we require [19]. These are directly given by $\mathbf{V}_a^\Phi = \Phi(\mathbf{X})\bar{\Gamma}_a$ and $\mathbf{V}_b^\Phi = \Phi(\mathbf{X})\bar{\Gamma}_b$, with $\bar{\Gamma}_a = \left(\bar{\gamma}_{a_1}, \ldots, \bar{\gamma}_{a_{p_\mathbf{a}}}\right)$ and $\bar{\Gamma}_b = \left(\bar{\gamma}_{b_1}, \ldots, \bar{\gamma}_{b_{p_\mathbf{b}}}\right)$.

Using this result in (9), we have

$$\begin{aligned}
\mathbf{A}^{\Phi^{-1}}\mathbf{B}^\Phi &= \sum_{i=1}^{p_\mathbf{b}}\sum_{j=1}^{p_\mathbf{a}} \frac{\lambda_{b_i}^\phi}{\lambda_{a_j}^\phi}\bar{\gamma}_{a_j}^T\Phi(\mathbf{X})^T\Phi(\mathbf{X})\bar{\gamma}_{b_i}\Phi(\mathbf{X})\bar{\gamma}_{a_j}\bar{\gamma}_{b_j}\Phi(\mathbf{X})^T \\
&= \Phi(\mathbf{X})\left(\sum_{i=1}^{p_\mathbf{b}}\sum_{j=1}^{p_\mathbf{a}} \frac{\lambda_{b_i}^\phi}{\lambda_{a_j}^\phi}\bar{\gamma}_{a_j}^T\mathbf{K}\bar{\gamma}_{b_i}\bar{\gamma}_{a_j}\bar{\gamma}_{b_i}^T\right)\Phi(\mathbf{X})^T,
\end{aligned}$$

Again, we can write $\mathbf{V}^\Phi = \Phi(\mathbf{X})\Gamma$. Therefore, we have

$$\begin{aligned}
\mathbf{A}^{\Phi^{-1}}\mathbf{B}^\Phi\mathbf{V}^\Phi &= \mathbf{V}^\Phi\Lambda^\Phi \\
\Phi(\mathbf{X})\left(\sum_{i=1}^{p_\mathbf{b}}\sum_{j=1}^{p_\mathbf{a}} \frac{\lambda_{b_i}^\phi}{\lambda_{a_j}^\phi}\bar{\gamma}_{a_j}^T\mathbf{K}\bar{\gamma}_{b_i}\bar{\gamma}_{a_j}\bar{\gamma}_{b_i}^T\right)\Phi(\mathbf{X})^T\Phi(\mathbf{X})\Gamma &= \Phi(\mathbf{X})\Gamma\Lambda^\Phi \\
\left(\sum_{i=1}^{p_\mathbf{b}}\sum_{j=1}^{p_\mathbf{a}} \frac{\lambda_{b_i}^\phi}{\lambda_{a_j}^\phi}\bar{\gamma}_{a_j}^T\mathbf{K}\bar{\gamma}_{b_i}\bar{\gamma}_{a_j}\bar{\gamma}_{b_i}^T\mathbf{K}\right)\Gamma &= \Gamma\Lambda^\Phi.
\end{aligned}$$

As we did previously in (4), we can now write the result $\left(\mathbf{A}^{\Phi^{-1}}\mathbf{B}^\Phi\right)_S$, which uses the set S of eigenvectors of $\mathbf{A}^\Phi$. In our equation above, this is

$$\left(\sum_{\gamma_{a_j}\in S}\sum_{i=1}^{p_\mathbf{b}} \frac{\lambda_{b_i}^\phi}{\lambda_{a_j}^\phi}\bar{\gamma}_{a_j}^T\mathbf{K}\bar{\gamma}_{b_i}\bar{\gamma}_{a_j}\bar{\gamma}_{b_i}^T\mathbf{K}\right)\Gamma = \Gamma\Lambda^\Phi. \tag{10}$$

This eigenvalue decomposition gives the basis vectors Γ, which directly provide the eigenvector $\mathbf{V}^\Phi$ of $\left(\mathbf{A}^{\Phi^{-1}}\mathbf{B}^\Phi\right)_S\mathbf{V}^\Phi = \mathbf{V}^\Phi\Lambda^\Phi$.

This is a desirable result, because $\bar{\gamma}_{a_j}^T\mathbf{K}\bar{\gamma}_{b_i}$ still acts as a switch, determining which of the vectors of $\mathbf{A}^\Phi$ and $\mathbf{B}^\Phi$ are correlated and by how much. This is made clear by the equalities $\mathbf{a}_j^\phi = \Phi(\mathbf{X})\bar{\gamma}_{a_j}^T$ and $\mathbf{b}_i^\phi = \Phi(\mathbf{X})\bar{\gamma}_{b_i}$, which show $\bar{\gamma}_{a_j}^T\mathbf{K}\bar{\gamma}_{b_i} = \mathbf{a}_j^{\phi^T}\mathbf{b}_i^\phi$. Hence, we can once more apply the correlation criterion derived in this paper in the kernel space, that is, in the solution derived in (10).

It is noticed that in (10),

$$tr\left(\sum_{i=1}^{p_{\mathbf{b}}}\sum_{j=1}^{p_{\mathbf{a}}}\frac{\lambda_{b_i}^{\phi}}{\lambda_{a_j}^{\phi}}\bar{\gamma}_{a_j}^T\mathbf{K}\bar{\gamma}_{b_i}\bar{\gamma}_{a_j}\bar{\gamma}_{b_i}^T\mathbf{K}\right)=\sum_{i=1}^{p_{\mathbf{b}}}\sum_{j=1}^{p_{\mathbf{a}}}\frac{\lambda_{b_i}^{\phi}}{\lambda_{a_j}^{\phi}}\left(\bar{\gamma}_{a_j}^T\mathbf{K}\bar{\gamma}_{b_i}\right)^2. \tag{11}$$

Similar to (3), we define the discriminant power of $\bar{\gamma}_{a_j}$ as

$$J(\bar{\gamma}_{a_j})=\sum_{i=1}^{p_{\mathbf{b}}}\frac{\lambda_{b_i}^{\phi}}{\lambda_{a_j}^{\phi}}\left(\bar{\gamma}_{a_j}^T\mathbf{K}\bar{\gamma}_{b_i}\right)^2,$$

which gives

$$tr\left(\sum_{i=1}^{p_{\mathbf{b}}}\sum_{j=1}^{p_{\mathbf{a}}}\frac{\lambda_{b_i}^{\phi}}{\lambda_{a_j}^{\phi}}\bar{\gamma}_{a_j}^T\mathbf{K}\bar{\gamma}_{b_i}\bar{\gamma}_{a_j}\bar{\gamma}_{b_i}^T\mathbf{K}\right)=\sum_{j=1}^{p_{\mathbf{a}}}J(\bar{\gamma}_{a_j}).$$

As in the linear case, the selection of the eigenvector set S should not be carried out using the discriminant power measure. Instead, we should define a new version of our criterion applicable to the non-linear approach defined above. As already noticed, in the non-linear case, the correlation value between the two metrics is given by

$$I_j=\sum_{i=1}^{p_{\mathbf{b}}}\left(\bar{\gamma}_{a_j}^T\mathbf{K}\bar{\gamma}_{b_i}\right)^2. \tag{12}$$

It then follows that

$$\begin{aligned}
\sum_{j=1}^{p_{\mathbf{a}}}I_j &= \sum_{j=1}^{p_{\mathbf{a}}}\sum_{i=1}^{p_{\mathbf{b}}}\left(\bar{\gamma}_{a_j}^T\mathbf{K}\bar{\gamma}_{b_i}\right)^2\\
&= \sum_{i=1}^{p_{\mathbf{b}}}\sum_{j=1}^{p_{\mathbf{a}}}\left(\frac{\gamma_{a_j}^T\mathbf{K}\mathbf{W}\mathbf{K}\gamma_{b_i}}{\lambda_{b_i}\sqrt{\gamma_{b_i}^T\mathbf{K}\gamma_{b_i}}\sqrt{\gamma_{a_j}\mathbf{K}\gamma_{a_j}}}\right)^2\\
&= \sum_{i=1}^{p_{\mathbf{b}}}\sum_{j=1}^{p_{\mathbf{a}}}\frac{\gamma_{b_i}^T\mathbf{K}\gamma_{b_i}}{\gamma_{a_j}^T\mathbf{K}\gamma_{a_j}}\left(\gamma_{a_j}^T\gamma_{b_i}\right)^2\\
&= \sum_{i=1}^{p_{\mathbf{b}}}\frac{\gamma_{b_i}^T}{\gamma_{b_i}^T\mathbf{K}\gamma_{b_i}}\left[\sum_{j=1}^{p_{\mathbf{a}}}\left(\gamma_{a_j}^T\mathbf{K}\gamma_{a_j}\right)\gamma_{a_j}\gamma_{a_j}^T\right]\gamma_{b_i}\\
&= \sum_{i=1}^{p_{\mathbf{b}}}\frac{\gamma_{b_i}^T}{\gamma_{b_i}^T\mathbf{K}\gamma_{b_i}}\mathbf{K}\gamma_{b_i}=p_{\mathbf{b}}.
\end{aligned}$$

Then, similar to the non-kernel case defined in (7), the normalized correlation value is defined as

$$f_j=\frac{1}{p_{\mathbf{b}}}I_j=\frac{1}{p_{\mathbf{b}}}\sum_{i=1}^{p_{\mathbf{b}}}\left(\bar{\gamma}_{a_j}\mathbf{K}\bar{\gamma}_{b_i}\right)^2.$$

Therefore, for this non-linear implementation, we can use the same correlation approach of principal components selection defined in this paper.

5 Experimental Results

In our first set of experiments we use three different datasets. One of these is the ETH-80 database previously introduced. The other sets are: the AR-face database [16] and the Sitting Posture Distribution Maps (SPDM) set of [21].

In the ETH test, we selected the 41 images of one randomly chosen object to represent the testing set. This provides a total of 328 test images. We used the rest of the images for training. In Fig. 3(a) we show the correlation plot of the eigenvectors of the sample covariance matrix with those of the between-class scatter matrix together with their approximation g_y. The vertical axis in Fig. 3(d) shows the recognition rate obtained when one keeps the first k eigenvectors shown in (a). The dotted lines in (a) and (d) represents the cutoff at the 90% confidence interval.

In Fig. 3(g) we show the results obtained when one selects the k eigenvectors that keep $r\%$ of the total variance. Here, we have included a set of large dots to indicate the recognition rates obtained when $r = \{90, 91, \ldots, 99\}$. Similarly, the dots in Fig. 3(j) specify the recognition rates when $r\%$ of the total discriminant information defined by $J(\mathbf{a}_j)$ is kept, where now $r = \{50, 60, 70, 80, 90, 91, \ldots, 99\}$. We note that in both cases, the recognition rate vary considerably depending on the value of r. In (g) the results vary from a high of 75.9% to a low of 68.29%, which are achieved when $r = 96$ and $r = 90$, respectively. In (j) the recognition rates vary from 75% to 51.8%, and the maximum is achieved when $r = 60$. We will now show that the values of r that maximizes (or minimizes) the recognition rates obtained using these two methods, are very different when the database changes. However, our method will remain stable (that is, the results obtained with our method will generally be one of the top regardless of the database used).

To see this, we will use the images of the AR-face database. The AR-face set consists of frontal-view face images of 100 people. Here, we will use the first 13 images taken during a first session, including images with distinct facial composites, occlusions and illuminations, for training. The other 13 images (taken during a second session) are used for testing. The images are first cropped around the face part (to remove hair and background information) and resized to a standard image size of 29 by 21 pixels. This yields a 609-dimensional feature space. During the cropping process we also aligned all faces to have a common position for the eyes and mouth. Again, the sample covariance matrix and the between-class scatter matrix are used as metrics for $\mathbf{A}$ and $\mathbf{B}$. Fig. 3(b) plots the correlation values f_j and their approximation g_y. Fig. 3(e) plots the corresponding recognition rates. Our result is specified by the dotted line.

As we did above, we now compare the results obtained using our method to those calculated when $r\%$ of the total variance or the total discriminant information is kept. This is shown in Figs. 3(h) and (k). Note that the maximum recognition rate in (h) is obtained when $r = 99$ and in (k) when $r = 80$. These are different to the optimal values of r reported in (g) and (j). Nonetheless, we see that the proposed approach provides a stable way of selecting the best bases of $\mathbf{A}$ – only pruning those bases that are either noisy (reduce the recognition rate) or uninformative (do not provide a statistically significant increment of the recognition rate).

In the SPDM database, the sample feature vectors were collected using a chair equipped with a pressure sensor sheet. Feature vectors correspond to $1,280$ pressure (force) values given by equally distributed points on the seating-pan of the chair. Here, the goal is to classify each feature vector into one of the 10 possible sitting postures [21]. The SPDM database includes five samples for each of the postures for a total of 50 people. Of the five samples available from each person, three are randomly selected as training samples and two for testing. This divides the data to $1,500$ samples for training and $1,000$ for testing. Correlation values f_j and their approximation g_y are shown in Fig. 3(c). The

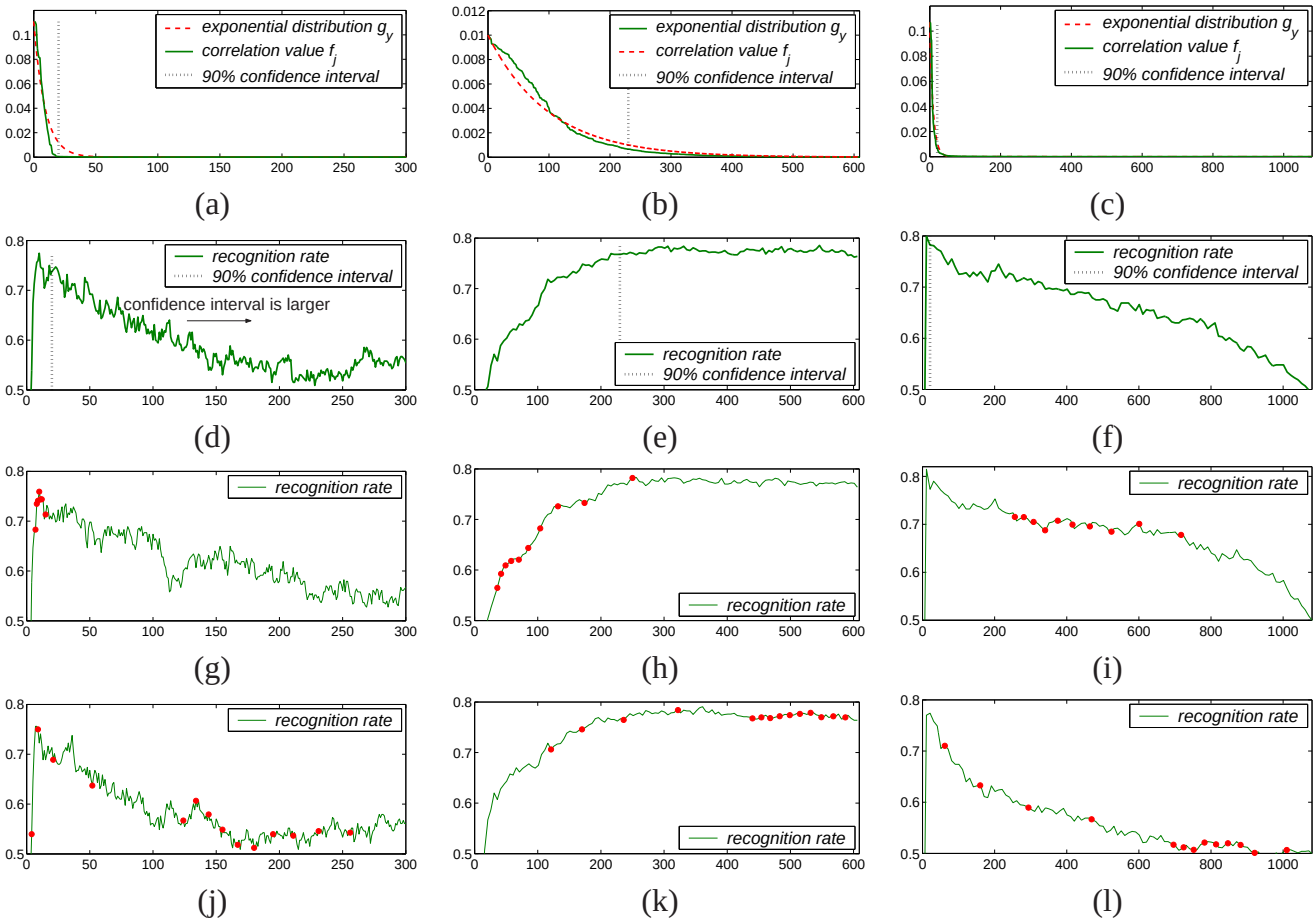

Figure 3: *In (a-c) we have ordered the eigenvectors of the covariance matrix* $\mathbf{A}$ *obtained from the ETH, AR and SPDM databases, from most to least correlated to* $ran(\mathbf{B})$*. The number of eigenvectors used is specified by the horizontal axis. In (d-f) We show the successful recognition rates when the* k *first eigenvectors, as ordered in (a-c), are used. Here, the dotted vertical line is the result given by our algorithm, and the vertical axis specifies the probability of correct classification. (g-i) Show the recognition rates obtained when* k *eigenvectors are kept and these are ordered in decreasing order of eigenvectors. (j-l) Do the same, but when the eigenvectors are ordered from most to least discriminant as given by* $J(\mathbf{a}_j)$*. The dots in (g-l) specify the successful classification rate when different percentages of the variance (or discriminant power) are retained.*

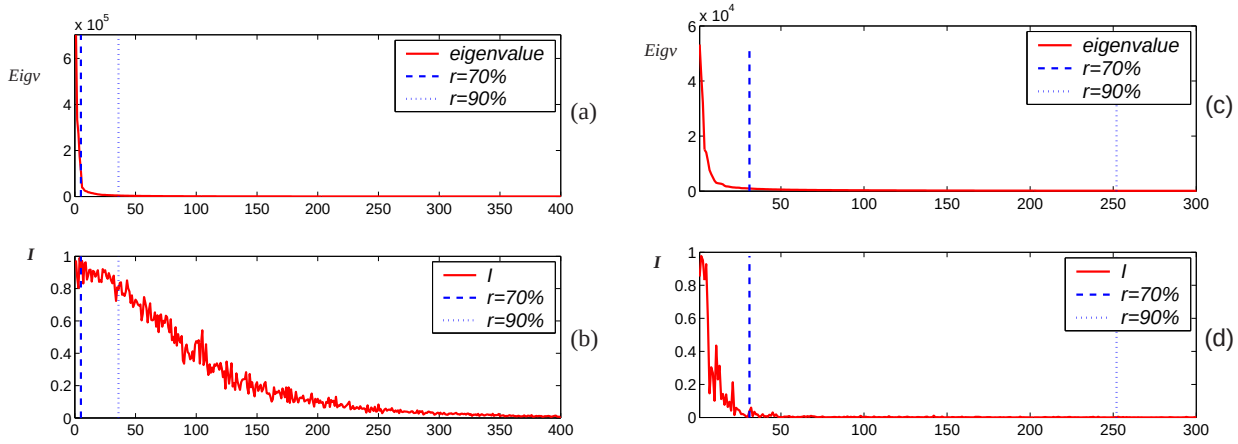

Figure 4: *Shown here are the eigenvalues of (a-b) AR and (c-d) SPDM datasets. In (a,c) the eigenvalues are ordered in decreasing order. (b,d) Shows the correlation between the eigenvectors of* $\mathbf{A}$ *and* $\mathbf{B}$ *– keeping the same order as in (a) and (c).*

corresponding recognition rates are in Fig. 3(f), where the dotted line marks our solution. In this case, the results obtained with the other two methods are unable to select a r which archives a good performance. This is shown in Figs. 3(i) and (l).

To see the difference of eigenvalue selection between our approach (presented in this paper) and the others tested above, we show how each algorithm sorts the principal components of the covariance matrix. The plots for the ETH database were given in Fig. 2(a). In Fig. 2(b) we demonstrated that some of the selected eigenvectors of $\mathbf{A}$ do not correlate with the solution space and, hence, cannot contribute to the solution. The problem is that some of these eigenvector represent data noise and keeping (other) unimportant eigenvectors may prevents us from choosing those that are useful in classification. This problem is resolved by selecting those principal components based on their correlation with the solution space as done by the proposed algorithm. This result is shown in Fig. 3(a).

We now do the same for the AR and SPDM datasets. In Fig. 4(a,c), we show the eigenvalues of the covariance matrix sorted in decreasing order. Fig. 4(b,d) illustrates that some of the corresponding eigenvectors of $\mathbf{A}$ are not correlated with the solution space and should not have been considered. This was resolved by our approach, whose results were shown in Fig. 3(b-c).

A statistical analysis of the results obtained using the ETH and SPDM databases is shown in the first two rows of Table 1. In this case, a bootstrap strategy is used. In the ETH set, the 41 images of a randomly selected object are used for testing and the rest for training. This is repeated 100 times. In the SPDM set 100 samples of each posture are randomly selected for training and the rest are used for testing. This is also repeated 100 times. Average and standard deviations are shown in the table.

In these results, we can see that the proposed algorithm consistently outperforms the others. This is because, in the ETH-80 and SPDM datasets, there is a considerable amount of noisy bases that make the classification problem challenging.

When the number of noisy features is reduced, this difference should become smaller. To test this, we used two additional databases from the UCI repository [3]. The first is the "Multi-feature Digit Dataset" (MDD), which includes 200 handwritten samples of each of the ten digits (0 to 9), and we used a representation (MDD-zer) based on 47 of the Zernike moments of the image [3]. The number of noisy bases in a 47-dimensional space is expected to be much lower than those of the ETH and SPDM sets. The bootstrap approach is employed to produce our results, with a training set containing 50%

Table 1: *Statistical comparison. In this table, eig_r means we use the eigenvectors that keep r% of the total variance, and $J(\mathbf{a})_r$ indicates that all those eigenvectors of $\mathbf{A}$ that keep r% of the total discriminant power given by $J(\mathbf{a})$ are used. In these experiments a bootstrap approach is used to generate average and standard deviations. Bolded results indicate statistical significance.*

Algorithm	Proposed Approach	eig_{70}	eig_{90}	$J(\mathbf{a})_{70}$	$J(\mathbf{a})_{90}$	Fisher-Rao's LDA
ETH80	**71.02** (4.39)	56.1 (4.02)	**72.56** (4.35)	66.83 (4.70)	61.99 (4.07)	59.93 (3.61)
SPDM	**75.53** (1.10)	**75.44** (1.12)	70.45 (1.26)	59.49 (1.48)	51.12 (1.52)	47.44 (1.51)
MDD-zer	**79.5** (1.02)	64.99 (1.09)	78.7 (1.07)	77.36 (1.38)	75.17 (3.81	78.09 (1.11)
Ionosphere	79 (3.01)	77.6 (3.14)	81.4 (2.40)	79.2 (3.33)	80.9 (2.57)	82.8 (2.80)

Table 2: *Results for the non-linear approach. In this table, eig_r means we use the eigenvectors that keep r% of the total variance, and $J(\gamma_a)_r$ indicates that all those eigenvectors of $\mathbf{M}_a$ that keep r% of the total discriminant power given by $J(\gamma_a)$ are used. In these experiments a bootstrap approach is used to generate average and standard deviations.*

Algorithm	Proposed Approach	eig_{70}	eig_{90}	$J(\gamma_a)_{70}$	$J(\gamma_a)_{90}$
ETH80	**72.87** (3.38)	40.21 (9.8)	57.9 (3.74)	52.84 (8.14)	64.76 (4.07)
SPDM	**73.93** (1.52)	23.79 (1.59)	55.2 (1.91)	64.98 (1.2)	73.53 (1.32)
MDD-zer	**77.21** (1.25)	43.58 (0.84)	60.98 (1.24)	60.15 (1.47)	76.22 (1.69)
Ionosphere	75.11 (3.99)	70.17 (3.85)	74.17 (3.71)	75.14 (3.58)	76.51 (3.06)

of the samples (randomly selected) and the testing set containing the remaining 50%. The operation is repeated 100 times and average and standard deviations are reported. As one can see in Table 1, the difference in performance is now reduced.

To further illustrate this point we used the Ionosphere set of [3], where the dimensionality of the data is even smaller. In this case, 34 radar attributes are employed to describe signals bouncing off the ionosphere. Our task is to determine whether the incoming signal is good (i.e., shows evidence of some type of structure) or not. Again half of the 351 samples are used for training and half for testing. This is repeated 100 times and average and standard deviations are shown in Table 1. As the dimensionality becomes more manageable (34 dimensions in this final test), the number of noisy bases decreases. As a consequence, and as predicted, Fisher-Rao's LDA and the other methods perform better, making it less necessary to use pruning methods.

As it is well known, the classification results obtained when the number of samples per class is small can be improved by adding a regularizing term to the sample covariance matrix [8, 15]. We have used this trick to boost the results obtained a bit further. Here, the results obtained with regularized LDA were: 71.89 (4.25) for ETH, 72.82 (1.12) for SPDM, 79.13 (1.05) for MDD-zer, and 80.14 (2.44) for Ionosphere. The results obtained with the regularized version of LDA presented in this paper were: 74.87 (4.25) for ETH, 76.05 (1.22) for SPDM, 79.39 (0.78) for MDD-zer, and 79.36 (4.71) for Ionosphere. As we can see from these results, the comparison of our approach with LDA parallels those presented in the tables.

Finally, we repeated our experiments using the non-linear extension of our method presented in

Section 4. In here we used the two metrics defined earlier for $\mathbf{M}_a$ and $\mathbf{M}_b$, and we selected the kernel function the RBF, defined as $k(\mathbf{x},\mathbf{y}) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\varsigma^2}\right)$, with ς the parameter to be estimated using the leave-one-out test on the training data.

To provide a statistical analysis of the results we used the bootstrap strategy described earlier for the results of Table 1. The difference is that we now used the kernel approach. These results are in Table 2. As we saw in the linear case, our method is superior to the other methods wherever the number of features is sufficiently large to require the pruning of the noisy bases. When the number of features is small, all the methods are more or less equivalent.

6 Conclusions

The classical formulation of LDA, based on the simultaneous diagonalization of two symmetric matrices (i.e., $\mathbf{A}^{-1}\mathbf{B}\mathbf{V} = \mathbf{V}\Lambda$), is known to be very sensitive to the noisy basis vectors in $\mathbf{A}$. In this paper, we have presented a simple correlation-based criterion that can be efficiently used to prune these noisy bases without eliminating those associated to the true discriminant power of the underlying structure of the data. Experimental results using five different databases have been used to illustrate the robustness of the proposed approach. These have demonstrated the superiority of the proposed approach (in pruning the noisy vectors) over a variety of known alternatives. We note that while existing algorithms only work in some particular cases, where the assumptions of the method hold, the proposed approach works well on all datasets tested. This is thanks to the underlying main idea of the paper which targets the problems associated to the generalized eigenvalue decomposition equation, rather than making assumptions on the type of the pdf. A non-linear extensions of the approach was also presented.

Acknowledgments

We thank the referees for their comments. We also thank Onur Hamsici for discussion. This research was partially supported by NIH under grant R01 DC 005241.

Appendix

Appendix A Notation

$\mathbf{A}$, $\mathbf{B}$	Matrices that define the metric to be minimized and maximized in the generalized eigenvalue decomposition $\mathbf{BV} = \mathbf{AV\Lambda}$.
$p_{\mathbf{a}}$	Rank of $\mathbf{A}, i.e.$,rank(**A**).
$\lambda_{\mathbf{a}_j}$, $\mathbf{a}_j$	Eigenvalues and eigenvectors of $\mathbf{A}$, $1 \leq j \leq p_{\mathbf{a}}$.
$p_{\mathbf{b}}$	Rank of $\mathbf{B}, i.e.$,rank(**B**).
$\lambda_{\mathbf{b}_i}$, $\mathbf{b}_i$	Eigenvalues and eigenvectors of $\mathbf{B}$, $1 \leq i \leq p_{\mathbf{b}}$.
$\phi(.)$	Mapping function that transforms a non-linearly separable problem in a linearly separable one.
$\mathbf{A}^{\Phi}$, $\mathbf{B}^{\Phi}$	Matrices that define the metric to be minimized and maximized in the space $\mathcal{F}$ given by the mapping function $\phi(.)$.
$\mathbf{a}_j^{\phi}$, $\mathbf{b}_i^{\phi}$	Basis vectors defined in the high-dimensional space $\mathcal{F}$.
γ_{a_j}, γ_{b_i}	Basis vectors of the metrics $\mathbf{M}_a$ and $\mathbf{M}_b$ used to define the kernel space.
$\mathbf{K}$	The Gramm matrix.
d	Dimensionality of the data.
S	A set of k elements drawn from $\{\mathbf{a}_1, \cdots, \mathbf{a}_{p_{\mathbf{a}}}\}$, without repetition.
$J(\mathbf{a}_j)$	The discriminant power of each $\mathbf{a}_j$, defined as $\frac{\mathbf{a}_j^T \mathbf{B} \mathbf{a}_j}{\lambda_{\mathbf{a}_j}}$.
r	The ratio representing $\frac{\sum_{j=1}^{k} \lambda_{\mathbf{a}_j}}{\sum_{j=1}^{p_{\mathbf{a}}} \lambda_{\mathbf{a}_j}}$ of the total variance or $\frac{\sum_{j=1}^{k} J(\mathbf{a}_j)}{\sum_{j=1}^{p_{\mathbf{a}}} J(\mathbf{a}_j)}$ of the total discriminant power.
$ran(\mathbf{M})$	Range space of $\mathbf{M}$.
I_j	Correlation value of each $\mathbf{a}_j$ to the $ran(\mathbf{B})$.
f_j	Normalized correlation value of each $\mathbf{a}_j$ to the $ran(\mathbf{B})$.
g_y	Exponential density function $g_y = \lambda e^{-\lambda y}$.
h	The probability of confidence interval of g_y. In this paper, $h = .9$.

Appendix B

It is interesting to further note that there is a direct relation between the subspace generated by those few eigenvectors of (4) that are associated to the largest eigenvalues and the subspace obtained with the classical PCA-LDA algorithm. This can be stated in the following result [23].

Result 1. *The projection matrix* $\mathbf{V}$ *found by (5) and the projection matrix* $\mathcal{A}_k\tilde{\mathbf{V}}$ *given by the following generalized eigenvalue decomposition*

$$\left(\mathcal{A}_k^T \mathbf{B} \mathcal{A}_k\right) \tilde{\mathbf{V}} = \left(\mathcal{A}_k^T \mathbf{A} \mathcal{A}_k\right) \tilde{\mathbf{V}} \tilde{\Lambda},$$

are identical, where the matrix $\mathcal{A}_k$ *has* k *columns, which are the* k *eigenvectors of* $\mathbf{A}$ *included in the set of* S *(Eq. (4)).*

Proof. The eigenvalue decomposition of $\mathbf{A}$ and $\mathbf{B}$ can be written as:

$$\mathbf{A} = \sum_{j=1}^{p_{\mathbf{a}}} \lambda_{\mathbf{a}_i} \mathbf{a}_i \mathbf{a}_i^T = \mathcal{A} \Lambda_{\mathbf{a}} \mathcal{A}^T, \qquad \mathbf{B} = \sum_{i=1}^{p_{\mathbf{b}}} \lambda_{\mathbf{b}_i} \mathbf{b}_i \mathbf{b}_i^T,$$

where $\mathcal{A} = [\mathbf{a}_1, \cdots, \mathbf{a}_{p_{\mathbf{a}}}] = [\mathcal{A}_k, \mathcal{A}_{p_{\mathbf{a}}-k}]$, where $\mathcal{A}_k$ contains k columns of $\mathcal{A}$ included in the set of S (Eq. (4)) and $\mathcal{A}_{p_{\mathbf{a}}-k}$ contains the remaining $p_{\mathbf{a}} - k$ eigenvectors of $\mathbf{A}$. We first project the data onto

the subspace spanned by $\mathcal{A}_k$. If in the reduced space, the two metric matrices corresponding to $\mathbf{A}$ and $\mathbf{B}$ are

$$\begin{aligned}\tilde{\mathbf{B}} &= \mathcal{A}_k^T \mathbf{B} \mathcal{A}_k, \\ \tilde{\mathbf{A}} &= \mathcal{A}_k^T \mathbf{A} \mathcal{A}_k = \Lambda_{\mathcal{A}_\mathbf{k}},\end{aligned}$$

the generalized eigenvalue decomposition of $\tilde{\mathbf{B}}$ with respect to $\tilde{\mathbf{A}}$ is:

$$\begin{aligned}\tilde{\mathbf{B}}\tilde{\mathbf{V}} &= \tilde{\mathbf{A}}\tilde{\mathbf{V}}\tilde{\Lambda} \\ \mathcal{A}_k^T \mathbf{B} \mathcal{A}_k \tilde{\mathbf{V}} &= \Lambda_{\mathcal{A}_k} \tilde{\mathbf{V}} \tilde{\Lambda}. \end{aligned} \tag{9}$$

where, the diagonal matrix $\Lambda_{\mathcal{A}_k}$ represents the eigenvalues of the k eigenvectors in $\mathcal{A}_k$.

Using the knowledge of basic linear algebra, (9) can be rewritten as:

$$\begin{aligned}\Lambda_{\mathcal{A}_k}^{-1} \mathcal{A}_k^T \mathbf{B} \mathcal{A}_k \tilde{\mathbf{V}} &= \tilde{\mathbf{V}}\tilde{\Lambda} \\ \mathcal{A}_k \Lambda_{\mathcal{A}_k}^{-1} \mathcal{A}_k^T \mathbf{B} \mathcal{A}_k \tilde{\mathbf{V}} &= \mathcal{A}_k \tilde{\mathbf{V}}\tilde{\Lambda} \\ \begin{pmatrix}\mathbf{a}_1, & \ldots, & \mathbf{a}_k\end{pmatrix} \begin{pmatrix}\frac{1}{\lambda_{\mathbf{a}_1}} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{\lambda_{\mathbf{a}_k}}\end{pmatrix} \begin{pmatrix}\mathbf{a}_1^T \\ \vdots \\ \mathbf{a}_k^T\end{pmatrix} \sum_{i=1}^{p_\mathbf{b}} \lambda_{\mathbf{b}_i} \mathbf{b}_i \mathbf{b}_i^T \mathcal{A}_k \tilde{\mathbf{V}} &= \mathcal{A}_k \tilde{\mathbf{V}}\tilde{\Lambda} \\ \begin{pmatrix}\frac{1}{\lambda_{\mathbf{a}_1}}\mathbf{a}_1 & \cdots & \frac{1}{\lambda_{\mathbf{a}_k}}\mathbf{a}_k\end{pmatrix} \begin{pmatrix}\sum_{i=1}^{p_\mathbf{b}} \lambda_{\mathbf{b}_i} \mathbf{a}_1^T \mathbf{b}_i \mathbf{b}_i^T \\ \vdots \\ \sum_{i=1}^{p_\mathbf{b}} \lambda_{b_i} \mathbf{a}_k^T \mathbf{b}_i \mathbf{b}_i^T\end{pmatrix} \mathcal{A}_k \tilde{\mathbf{V}} &= \mathcal{A}_k \tilde{\mathbf{V}}\tilde{\Lambda} \\ \sum_{j=1}^{k} \sum_{i=1}^{p_\mathbf{b}} \frac{\lambda_{\mathbf{b}_i}}{\lambda_{\mathbf{a}_j}} \langle \mathbf{a}_j, \mathbf{b}_i \rangle \mathbf{a}_j \mathbf{b}_i^T \mathcal{A}_k \tilde{\mathbf{V}} &= \mathcal{A}_k \tilde{\mathbf{V}}\tilde{\Lambda} \\ \sum_{\mathbf{a}_j \in S} \sum_{i=1}^{p_\mathbf{b}} \frac{\lambda_{\mathbf{b}_i}}{\lambda_{\mathbf{a}_j}} \langle \mathbf{a}_j, \mathbf{b}_i \rangle \mathbf{a}_j \mathbf{b}_i^T \mathcal{A}_k \tilde{\mathbf{V}} &= \mathcal{A}_k \tilde{\mathbf{V}}\tilde{\Lambda} \end{aligned} \tag{10}$$

Compare (10) and (4), we know that $\mathbf{V}$ is identical to $\mathcal{A}_k \tilde{\mathbf{V}}$. □

References

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

[2] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.

[3] C.L. Blake and C.J. Merz. *UCI Repository of Machine Learning Databases*. `http://www.ics.uci.edu/~mlearn/MLRepository.html`.

[4] P.A. Devijver and J Kittler. *Pattern Recognition: A statistical approach*. Prentice Hall, 1982.

[5] M.J. Er, W. Chen, and S. Wu. High-speed face recognition based on discrete cosine transform and rbf neural networks. *IEEE Trans. on Neural Networks*, 16(3):679–691, 2005.

[6] K. Etemad and R. Chellappa. Discriminant analysis for recognition of human face images. *Journal of the Optical Society of America A*, 14:1724–1733, 1997.

[7] R.A. Fisher. The statistical utilization of multiple measurements. *Annals of Eugenics*, 8:376–386, 1938.

[8] J.H. Friedman. Regularized discriminant analysis. *Journal of the American Statistical Association*, 84(405):165–175, 1989.

[9] K. Fukunaga. *Introduction to Statistical Pattern Recognition (*$2^{nd}Edition$*)*. Academic Press, 1990.

[10] I.T. Jolliffe. *Principle Component Analysis*. Springer, 2002.

[11] M. Kyperountas, A. Tefas, and I. Pitas. Weighted piecewise lda for solving the small sample size problem in face verification. *IEEE Trans. on Neural Networks*, 18(2):506–519, 2007.

[12] B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. In *In Proc. IEEE Computer Vision and Pattern Recognition*, pages II:409–415, 2003.

[13] J. Lu, K.N. Plataniotis, and A.N. Venetsanopoulos. Face recognition using kernel direct discriminant analysis algorithms. *IEEE Trans. on Neural Networks*, 14(1):117–126, 2003.

[14] J. Lu, K.N. Plataniotis, and A.N. Venetsanopoulos. Face recognition using LDA-based algorithms. *IEEE Trans. on Neural Networks*, 14(1):195–200, 2003.

[15] J. Lu, K.N. Plataniotis, and A.N. Venetsanopoulos. Regularization studies of linear discriminant analysis in small sample size scenarios with application to face recognition. *Pattern Recognition Letter*, 26(2):181–191, 2005.

[16] A.M. Martínez and A.C. Kak. PCA versus LDA. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(2):228–233, 2001.

[17] A.M. Martínez and M. Zhu. Where are linear feature extraction methods applicable? *IEEE Trans on Pattern Analysis and Machine Intelligence*, 27(12):1934–1944, 2005.

[18] C.R. Rao. The utilization of multiple measurements in problems of biological classification. *J. Royal Statistical Soc., B*, 10:159–203, 1948.

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

[20] D.L. Swets and J.J. Weng. Using discriminant eigenfeatures for image retrieval. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(8):831–836, 1996.

[21] H.Z. Tan, L.A. Slivovsky, and A. Pertland. A sensing chair using pressure distribution sensors. *IEEE Trans. Mechatronics*, (3):261–268, 2001.

[22] H. Yu and H. Yang. A direct lda algorithm for high-dimensional data - with application to face recognition. *Pattern Recognition*, 34(10):2067–2070, 2001.

[23] M. Zhu and A.M. Martínez. Selecting principal components in a two-stage LDA algorithm. *In Proc. IEEE Computer Vision and Pattern Recognition, New York*, 1:132–137, 2006.

[24] M. Zhu and A.M. Martínez. Subclass discriminant analysis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 28(8):1274–1286, 2006.