Bilinear Analysis for Kernel Selection and Nonlinear Feature Extraction

Shu Yang, Shuicheng Yan, *Member, IEEE*, Chao Zhang, and Xiaoou Tang, *Senior Member, IEEE*

***Abstract*—This paper presents a unified criterion, Fisher + kernel criterion (FKC), for feature extraction and recognition. This new criterion is intended to extract the most discriminant features in different nonlinear spaces, and then, fuse these features under a unified measurement. Thus, FKC can simultaneously achieve nonlinear discriminant analysis and kernel selection. In addition, we present an efficient algorithm Fisher + kernel analysis (FKA), which utilizes the bilinear analysis, to optimize the new criterion. This FKA algorithm can alleviate the ill-posed problem existed in traditional kernel discriminant analysis (KDA), and usually, has no singularity problem. The effectiveness of our proposed algorithm is validated by a series of face-recognition experiments on several different databases.**

***Index Terms*—Bilinear analysis, discriminant analysis, face recognition, feature extraction, Fisher criterion, kernel selection.**

I. Introduction

FISHER criterion has been widely used in the field of pattern recognition [37]. Recently, kernel-based methods have been introduced in Fisher criterion to form various kernel Fisher methods [1], [7], [20], [21], [24], [44]. These methods are usually called kernel Fisher discriminant analysis (KFD) or kernel discriminant analysis (KDA) [28]. Generally speaking, KDA first maps the original data into a higher dimensional feature space via a nonlinear mapping, and then, applies linear discriminant analysis (LDA) [8] in this higher dimensional feature space. In KDA, we do not need to know explicitly the nonlinear mapping; the kernel function, i.e., inner product of the data pairs in feature space, is enough to derive final solution [29], [31]. Due to the capability of being able to extract the most discriminant features [26] from the nonlinear data and the feasibility in computation, KDA and its refined versions have been widely applied in computer vision, especially for face-recognition tasks [18], [19], [39].

However, in real-world applications, these KDA algorithms always encounter two following problems: 1) the ill-posed problem [22], [14], [41] and 2) the selection of the kernel function under a given data set [4], [15]. Moreover, these two problems usually interwind, which greatly limits the effectiveness of KDA.

For the ill-posed problem, there have been a number of regularization techniques and matrix-based methods that might alleviate the problem [1], [40], [20], [22], [19], [39]. While these methods can alleviate or even solve the singular problem in theory, they still cannot be successfully integrated into the process of kernel selection.

In addition, the problem of kernel selection itself does not have a satisfactory solution yet. In general, the methods for finding the optimal kernel [38] can be classified into two categories: one is independent of the subsequent learning algorithm and the other is not. For the first category, new kernels are the refined ones of the traditional kernels with special motivations, such as Bhattacharyya point set kernel in [11] and the cosine kernel in [17]. These methods are not always effective for specific problems in practice. For the second category, such as boosting kernel for support vector machine (SVM) [3] and radial basis function (RBF) kernel parameter selection for KDA [10], the procedure for kernel selection is often time consuming and usually restricted to a certain kind of kernels.

Due to these two problems, it is even more difficult to conduct kernel selection and at the same time to prevent the ill-posed problem. As discussed later in this paper, we prove that the ill-posed problem of KDA brings about a special form of overfitting, in which the data of the same class are mapped onto the same point. Under this condition, traditional KDA cannot tell which kernel is better, so it is not feasible to conduct kernel selection using the traditional Fisher criterion. To address such an issue, there are some methods which carry out kernel selection after adding a regularization term [6], [12]. Though some of the methods [12] provide convex optimization solution, there is still no method that can find an optimal regularization term and optimal kernel simultaneously.

Based on the previous considerations, we try to find a new criterion which helps overcome the two disadvantages of KDA algorithm. However, our aim is not at finding an optimal parameter for the kernel functions or selecting a specific kernel function from a function set. Denote the set constructed by both the training and testing samples as $\{x_1, x_2, \ldots, x_t\}$. In KDA and many other kernel related classification algorithms, we only need the inner product $\phi(x_i) \cdot \phi(x_j)$ between every two samples x_i and $x_j (i, j = 1, \ldots, t)$ in the higher dimensional feature space. According to the kernel trick [29], we have the kernel matrix $K_{i,j} = K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$, which is also known

Manuscript received September 23, 2005; revised May 12, 2006 and December 6, 2006; accepted December 18, 2006. This work was supported in part by the Grants NSFC 60572043, NKBRPC 2004CB318005, FANEDD 200038, and N_CUHK 409/03.

S. Yang is with the Department of Mathematics and Statistics, Boston University, Boston, MA 02215 USA (e-mail: sophyshu@gmail.com).

S. Yan is with the Beckman Institute, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA (e-mail: scyan@ifp.uiuc.edu).

C. Zhang is with the National Laboratory on Machine Perception, Peking University, Beijing 100871, China (e-mail: chzhang@cis.pku.edu.cn).

X. Tang is with the Visual Computing Group, Microsoft Research Asia, Beijing, China and with the Department of Information Engineering, the Chinese University of Hong Kong, Shatin, New Territories, Hong Kong (e-mail: xitang@microsoft.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNN.2007.894042

as Gram matrix [29]. Therefore, what we need is a final kernel matrix K, with the element of $K_{i,j}, (i, j = 1, \ldots, t)$, which is optimal for a given data set. Besides, we realize that an optimal combination of various kernels [5], [6] may be a proper way for kernel selection. Therefore, we propose a novel criterion, called Fisher $+$ kernel criterion (FKC) for discriminant analysis, in which kernel selection and nonlinear discriminant analysis problem are simultaneously solved by optimizing a single objective function. Specifically, each sample of the original input data is mapped to a feature matrix through a compound map, which is constructed by combining several different nonlinear mappings. Then, with the new criterion, the feature matrix is projected into a lower dimensional space through a left- and a right-projection space; and thus, we find a projection matrix as the final kernel matrix. This projection matrix is optimal under the overall consideration of the classification ability within various kernel spaces. Accordingly, we propose an iterative optimization procedure Fisher $+$ kernel analysis (FKA) for obtaining the solution to FKC.

The remainder of this paper is organized as follows. An overview of the KDA algorithm is provided in Section II. In Section III, we analyze the ill-posed problem existed in KDA and present a special form of overfitting along with the constrained solution as a counter method to this ill-posed problem. FKC and its optimization procedure FKA are presented in Section IV. Validation experiments and conclusions are presented in Sections V and VI.

II. Review of Kernel Discriminant Analysis

Assume that we are given a collection of training image samples denoted as $\chi = \{x_1, x_2, \ldots, x_n\}$, $x_i \in \Re^m$; the sample x_i belongs to the l_ith class, where $l_i \in \{1, 2, \ldots, p\}$, and n_{l_i} is the number of the samples belonging to the l_ith class.

Given a nonlinear mapping ϕ, the original input space χ is mapped into a higher dimensional feature space $\mathcal{F}$[1]

$$\begin{aligned} \phi : \chi &\to \mathcal{F} \\ x &\mapsto \phi(x). \end{aligned}$$

Define the inner product in the feature space as $k(x, y) = \phi(x) \cdot \phi(y)$, where $k(x, y)$ is known as a kernel function.

Taking $\phi(x)$ to be an ordinary vector, we define scatter matrices of the samples in $\mathcal{F}$ as follows. The intraclass scatter matrix S_W and the interclass scatter matrix S_B are

$$S_B = \sum_{l=1}^{p} n_l (\bar{\phi}^l - \bar{\phi})(\bar{\phi}^l - \bar{\phi})^T \quad (1)$$

$$S_W = \sum_{i=1}^{n} (\phi(x_i) - \bar{\phi}^{l_i})(\phi(x_i) - \bar{\phi}^{l_i})^T \quad (2)$$

where $\bar{\phi}^l$ is the mean of the mapped samples belonging to the lth class, n_l is the number of samples in lth class, and $\bar{\phi}$ is the mean of all mapped samples. Similar to the scatter matrices in LDA, S_B measures the dispersion of the samples from different classes in the feature space $\mathcal{F}$, while S_W denotes the dispersion of the samples within the same class.

Using the scatter matrices, the main idea of KDA is to apply the Fisher criterion in the higher dimensional feature space $\mathcal{F}$. The criterion for KDA algorithm can be given as follows.

Criterion 1:

$$\psi^* = \arg\max_{\psi} \frac{\psi^T S_B \psi}{\psi^T S_W \psi}$$

where A^T with superscript T means the transposition of matrix $A, \psi \in \operatorname{span}\{\phi(x_i), i = 1, \ldots, n\} \subset \mathcal{F}$ and $\psi = \sum_{i=1}^{n} \alpha_i \phi(x_i)$.

The value of ϕ is usually unknown and ψ is a vector in form whose dimension may be infinite. Therefore, we rewrite the KDA's criterion with new representation of the scatter matrices and solve it by using the kernel trick [29]. Denote $\Phi = (\phi(x_1), \ldots, \phi(x_n))$. As shown in the Appendix I, we can define the scatter matrices in the feature space $\mathcal{F}$ directly as

$$S_W = \Phi^T M_W \Phi \quad \text{and} \quad S_B = \Phi^T M_B \Phi \quad (3)$$

where M_W and M_B are two constant matrices which only depend on the label of the data. With the form of the scatter matrices in (3), we can see that the information in S_W and S_B is separated into two groups: The distribution information of the data is contained in Φ while the class information of the label is represented by the matrices M_W and M_B.

Recalling that the elements of kernel matrix K are defined as $K_{i,j} = \phi(x_i) \cdot \phi(x_j)$, we denote the kernel matrix $K = \Phi^T \Phi$. Then, Criterion 1 can be rewritten as

$$\alpha^* = \arg\max_{\alpha} \frac{\alpha^T K^T M_B K \alpha}{\alpha^T K^T M_W K \alpha}$$

where $\alpha = (\alpha_1, \ldots, \alpha_n)^T$. Similar with the solution of traditional discriminant analysis [8], the optimal solution of Criterion 1 can be obtained from the generalized eigenvalue decomposition [8], and thus, we have

$$K^T M_B K \alpha = \lambda K^T M_W K \alpha. \quad (4)$$

If $K^T M_W K$ is invertible, (4) can be solved directly and Fisher criterion can be directly employed to extract the optimal discriminant features in the feature space. However, as shown in the Appendix II, the matrix M_W does not have full rank which makes $K^T M_W K$ not invertible. Thus, there always exists a vector satisfying $K^T M_W K \alpha = 0$, which means there is a vector ψ such that $\psi^T S_W \psi = 0$. Thus, the optimal problem for Criterion 1 is an ill-posed one under this condition. In Section III, we present a special form of overfitting resulting from this ill-posed problem. We also propose a regularization method as a solution to this ill-posed problem.

III. Overfitting in KDA and Counter Method

Notice that $\operatorname{rank}(K^T M_W K) \leq \operatorname{rank}(M_W) = n - p$, and hence, the dimension of the null space of $K^T M_W K$ is no less than p. Define α_∞ as the vectors mentioned in Criterion 1 which satisfy $\max_\psi(|\psi^T S_B \psi| / |\psi^T S_W \psi|) = +\infty$. The number of these α_∞ is at least p. With α_∞, the scatter distance of the same

[1] Actually, $\mathcal{F}$ is a so called reproducing kernel Hilbert space.

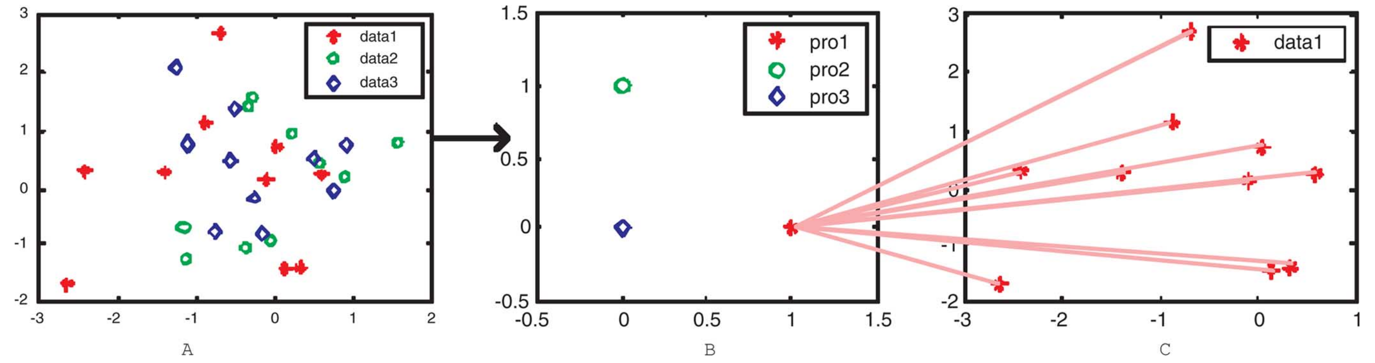


Fig. 1. Illustration of overfitting with three-classes toy data. (a) Original data of three classes in 2-D space. (b) The transformed results with KDA. (c) The corresponding original data for the first class.

class after being projected onto that direction is zero, i.e., all the samples from the same class are mapped to a same point in the projection dimension. We prove rigorously in the Appendix II how this can introduce a special form of overfitting. To make it easier to understand here, we use a toy data with three classes to illustrate this special overfitting in Fig. 1. With the projection direction calculated in the Appendix II, the original data of the same class are mapped onto the same point in the projection space.

Generally, various regularization methods are adopted to overcome this ill-posed problem. The most popular regularization method [22] is to use a penalized term $K^T M_W K + \lambda I$ instead of $K^T M_W K$ in (4), where I is the identity matrix. Although this method can solve the ill-posed problem in theory, regularization in this way also brings an extra parameter λ, which is rarely known as *a priori*, to be optimized. If we use the previous regularization, we first need to optimize the parameter λ. Denote the optimal parameter as λ_{opt}. We select a kernel function $k_{\lambda_{\text{opt}}}$, which only guarantees to be optimal under this λ_{opt}. However, since we aim to process the kernel selection and avoid the overfitting problem simultaneously, we may not get the best classification result if we choose to introduce the penalized term first. Therefore, in our paper we do not adopt such regularization in our kernel selection process.

According to Tikhonov's theory [32], we can restrict the solution in a constrained subspace of the whole solution space so as to obtain a stable solution as in the aforementioned penalized method. Therefore, in our algorithm, we apply the idea of constrained solution space to alleviate the ill-posed problem and help select a good kernel matrix. Admittedly, it is difficult to find a general constrained method for all the learning algorithms. We can still design a constrained solution space for a specific algorithm, such as for the KDA. Note that in the LDA, this space can be simply found by constraining the projection space, e.g., using the centroid space as the solution space [8], [9]. Similarly, we could design a constrained projection space to obtain a stable solution in KDA. This idea is adopted in our FKA to find a stable solution for the FKC introduced later.

IV. FKC and Optimization

From the kernel trick [29] and the definition of kernel function $k(x,y) = \phi(x) \cdot \phi(y)$, we can see that the kernel function is determined by the nonlinear mapping ϕ. Moreover, for the kernel matrix $K_{i,j} = k(x_i, x_j)$ to be a valid kernel function, it should satisfy the Mercer conditions [33]. Since $\phi{:}\chi \rightarrow \mathcal{F}$ and $\mathcal{F}$ is a reproducing kernel Hilbert space, the kernel matrix K is also decided by the inner product of the reproducing kernel Hilbert space $\mathcal{F}$.

In traditional methods, kernel selection is actually processed in $\mathcal{F}$. To combine the merits of various kernel functions, we define a new reproducing kernel Hilbert space. Denote a collection of nonlinear mapping functions as $\phi_j{:}\chi \rightarrow \mathcal{H}_j$, where $j \in \{1, \ldots, f\}$ and each $\mathcal{H}_j$ is a reproducing kernel Hilbert space. Construct a compound mapping $\hat{\phi}(x)$ as $\hat{\phi}(x) = (\phi_1(x)^T, \ldots, \phi_f(x)^T)^T$ such that $\hat{\phi}{:}\chi \rightarrow \mathcal{H}$. Using a proper definition of inner product, we prove in the Appendix III that $\mathcal{H}$ is a reproducing Hilbert space.

Our criterion is based on the whole reproducing kernel Hilbert space $\mathcal{H}$, so the inner product should be defined between any two vectors in $\mathcal{H}$ as a similarity measure. For simplicity, we assume that the feature vectors mapped by different $\hat{\phi}_j$ are independent, that is

$$\hat{\phi}_j(x) \cdot \hat{\phi}_j(y) = \hat{k}_j(x,y)$$

and

$$\hat{\phi}_i(x) \cdot \hat{\phi}_j(y) = 0 \quad (i \neq j). \tag{5}$$

Specifically, we define

$$\hat{\phi}_j = (\overbrace{0^T \cdots 0^T}^{1,\ldots,j-1}, \phi_j^T, \overbrace{0^T \cdots 0^T}^{j+1,\ldots,f})^T. \tag{6}$$

Similar to the case of $\hat{\phi}$ mentioned previously, we have that $\hat{\phi}_j{:}\chi \rightarrow \mathcal{H}$.

For each kernel function k_j, there exists a function ϕ_j such that $k_j(x,y) = \phi_j(x) \cdot \phi_j(y)$, which implies that the selection of a nonlinear mapping ϕ_j is equivalent to the selection of the kernel function k_j. Note that in our new reproducing kernel Hilbert space $\mathcal{H}$, every element $\hat{\phi}_j$ is constructed from the nonlinear mapping $\phi_j, j = \{1, \ldots, f\}$. Therefore, the task of extracting different feature vectors in $\mathcal{H}$ is equivalent to finding a good combination of the kernel functions.

A. FKC

First, we map the ith original sample x_i to a so-called feature matrix $\Phi(x_i) = [\hat{\phi}_1(x_i), \ldots, \hat{\phi}_f(x_i)], i = 1, \ldots, n$. It is clear that $\Phi(x_i)$ is constructed by vectors from $\mathcal{H}$. Then, for all the feature matrices $\Phi(x_j)$, we try to find at the same time an optimal combination of the different nonlinear feature $\hat{\phi}_i$ and good projection directions such that the distances of samples in the same class are minimized while the distances of samples between different classes are large after the projection, which is just the Fisher discriminant criterion.

The previous optimal combination $\hat{\phi}_j$ is achieved through multiplying the feature matrix $\Phi(x_i)$ by a right matrix V; then, the projection space is a subspace of the linear space spanned by $\hat{\phi}_j(x_i), j = 1, \ldots, f; i = 1, \ldots, n$. Denote by U the projection matrix; under the aforementioned projection directions, the projection of the feature matrix $\Phi(x_i)$ is $U^T \Phi(x_i) V$, called projection matrix $\Phi^P(x_i)$. Define

$$\|A\|_F^2 = \sum_{i,j} A_{ij}^2$$

as the Frobenius norm of the matrix A. Under this norm, the interclass distance D_B and the intraclass D_W for the projection of $\Phi(x_i)$ can be computed as

$$D_B = \sum_{c=1}^{p} n_c \|M_c - M\|_F^2$$

and

$$D_W = \sum_{i}^{n} \|\Phi^P(x_i) - M_{l_i}\|_F^2$$

where l_i denotes the class label of the ith projection matrix, M_c is the mean of the projection matrices from the class c, and M is the mean of all the projection matrices. Following Fisher criterion, the optimal projection matrices $\Phi^P(x_i), i = 1, \ldots, n$ satisfy the following: D_B, the distance between the projection matrices in the different classes, is as large as possible, while D_W, the distance within the projection matrices of the same class, is as small as possible. This process can be illustrated by the following optimization problem:

$$(U^*, V^*) = \arg\max_{U,V} \frac{D_B}{D_W}.$$

Replace D_B and D_W with the feature matrix $\Phi(x_i)$; we have our FKC as follows.

Criterion 2:

$$(U^*, V^*) = \arg\max_{U,V} \frac{\sum_l n_l \|U^T \bar{\Phi}_l V - U^T \bar{\Phi} V\|_F^2}{\sum_i \left\|U^T \Phi(x_i) V - U^T \bar{\Phi}_{l_i} V\right\|_F^2}$$

where $U \in \overline{\mathrm{span}\{\hat{\phi}_j(x_i) : i = 1, \ldots, n; j = 1, \ldots, f\}} = \mathcal{H}$ and $V \in \Re^{f \times f'}$ (f' is the final dimension given by the user), $\bar{\Phi}$ is the total average matrix of all the feature matrices $\Phi(x_i)$, and $\bar{\Phi}_l$ is the average matrix of the $\Phi(x_i)$ which belongs to lth class, and so $\bar{\Phi}_{l_i}$.

In Criterion 2, we use the right matrix V to search for a better combination of $\hat{\phi}_j$, $(j = 1, \ldots, f)$, hence, the right-projection space is a finite dimensional Euclidean space. In contrast, the left matrix U is used to find the most discriminative subspace in $\mathcal{H}$; therefore, the left-projection space is a subspace of $\mathcal{H}$.

B. Two Algorithms

Conventionally, we can search the left-projection subspace of $\mathrm{span}\{\hat{\phi}_j(x_i) : i = 1, \ldots, n; j = 1, \ldots, f\}$. However, a comprehensive search will result in a very large scale optimization problem. To cut the calculational overhead and present a regularized solution to the optimal problem, we propose two methods to constrain the left-projection space, which lead to the following two algorithms, FKA01 and FKA02, respectively. As discussed in Section III, we could also avoid the ill-posed problem by using this kind of constrained solutions, which will not bring about extra regularization parameter for selection.

FKA01: Assume that the left-projection space is constrained in $\mathrm{span}\{\tilde{\phi}(x_i) : i = 1, \ldots, n\}$, where $\tilde{\phi}(x_i) = \sum_{j=1}^{f} \hat{\phi}_j(x_i)$. Denote

$$U = [\tilde{\phi}(x_1), \tilde{\phi}(x_2), \ldots, \tilde{\phi}(x_n)] \cdot L \triangleq \tilde{\Phi} \cdot L, \qquad L \in \Re^{n \times q}.$$

Suppose that the projection of $\Phi(x_i)$ in left-projection space is K_i; then, the element of K_i, i.e., $K_i(a, b)$, has the form

$$K_i(a, b) = \tilde{\phi}(x_a) \cdot \hat{\phi}_b(x_i) = \phi_b(x_a) \cdot \phi_b(x_i) = k^b(x_a, x_i).$$

Different types of kernel functions k^b are chosen from the kernel bank $\{k^j, j = 1, \ldots, f\}$, which can be obtained from prior knowledge or simply defined by users. Basically, the larger is the kernel bank, the heavier the computational cost, yet the greater the potential for algorithmic classification capability. Hence, the selection of the kernel bank depends on the balance of computational cost and potential algorithmic classification capability.

FKA02: As discussed in Section III, we could also use the mapping of class centroids to approximate $\mathrm{span}\{\hat{\phi}_j(x_i) : i = 1, \ldots, n; j = 1, \ldots, f\}$. Then, the left-projection space is the constrained space of $\mathrm{span}\{\hat{\phi}_j(\bar{x}_l) : j = 1, \ldots, f; l = 1, \ldots, p\}$ and $\bar{x}_l$ is the centroid of lth class. Then, we define

$$U = [\Phi(\bar{x}_1), \Phi(\bar{x}_2), \ldots, \Phi(\bar{x}_p)] \cdot L \triangleq \tilde{\Phi} \cdot L, \qquad L \in \Re^{fp \times q}.$$

Denote the projection of $\Phi(x_i)$ in the left-projection space as K_i; then, the element of K_i, i.e., $K_i(a, b)$, has the form

$$\begin{aligned} K_i(a, b) &= \hat{\phi}_{a_1}(\bar{x}_{a_0}) \cdot \hat{\phi}_b(\bar{x}_i) = \delta_{a_1,b} \cdot \phi_b(\bar{x}_{a_0}) \cdot \phi_b(x_i) \\ &= \delta_{a_1,b} \cdot k^b(\bar{x}_{a0}, x_i) \end{aligned}$$

where $a = (a_0 - 1)f + a_1$.

These two methods have different left-projection spaces, but both of them can be solved in the following form. Denote that

$$\begin{aligned} &\bar{K}_l = \tilde{\Phi}^T \cdot \bar{\Phi}_l \quad \bar{K} = \tilde{\Phi}^T \cdot \bar{\Phi} \\ &K_i = \tilde{\Phi}^T \cdot \Phi_i \quad \bar{K}_{l_i} = \tilde{\Phi}^T \cdot \bar{\Phi}_{l_i}. \end{aligned} \tag{7}$$

Then, the Criterion 2 can be simplified to Criterion 3.

```
Fisher + Kernel Analysis
Given input sample set $\{x_i \in \Re^m, i = 1, \cdots, n\}$, class labels $l_i \in \{1, 2, \cdots, p\}$ and the desired final dimensions $q$ and $f'$.
1. Construct the kernel matrix defined in (7);
2. Initiate $V_0 \in \Re^{f \times f'}$;
3. For $t = 1, 2, \cdots, t_n$, Do
    a). For given $V_{t-1}$, calculate the optional $L_t$ as in (10);
    b). For given $L_t$, calculate the optimal $V_t$ as in (11-12);
    c). If $\|L_t L_t^T - L_{t-1} L_{t-1}^T\|_F < \varepsilon$ (a small value), $\|V_t V_t^T - V_{t-1} V_{t-1}^T\|_F < \varepsilon$ and $t > 2$, go to step 4; else, continues.
4. Out put the projection $L = L_t$ and $V = V_t$.
```

Fig. 2. Procedure for FKA.

Criterion 3:

$$(L^*, V^*) = \arg\max_{L,V} \frac{\sum_l n_l \|L^T \bar{K}_l V - L^T \bar{K} V\|_F^2}{\sum_i \left\|L^T K_i V - L^T \bar{K}_{l_i} V\right\|_F^2}.$$

The kernel matrices defined in (7) can be directly computed from the kernel bank $\{k_j, j = 1, \ldots, f\}$. Thus, the solution of Criterion 3 can be obtained by using general matrix analysis techniques. In our paper, we use an iterative procedure to find the optimal solution for Criterion 3.

C. Iterative Optimization Procedure

We implement the optimization procedure in two steps. First, L-step is to find an optimal L matrix and V-step to obtain the optimal V matrix. In the L-step, we fix the right matrix V to get the optimal L. Then, in the V-step, we fix L as the one computed in the L-step, and calculate the optimal V. The two steps are iterated for a certain number of times until converged. The two steps are as follows.

1) L-step: For a given $V \in \Re^{f \times f'}$, the objective function of Criterion 3 can be rewritten as

$$L^* = \arg\max_{L} \frac{\sum_l n_l \|L^T \bar{K}_l V - L^T \bar{K} V\|_F^2}{\sum_i \left\|L^T K_i V - L^T \bar{K}_{l_i} V\right\|_F^2}.$$

Since $\|A\|_F^2 = \text{Trace}(AA^T)$, we have

$$\begin{aligned}\sum_l n_l \|L^T \bar{K}_l V - L^T \bar{K} V\|_F^2 &= \sum_l n_l (\text{Trace}(L^T (\bar{K}_l V - \bar{K} V)(\bar{K}_l V - \bar{K} V)^T L)) \\ &= \text{Trace}(L^T S_B^V L)\end{aligned}$$

where

$$S_B^V = \sum_l n_l (\bar{K}_l V - \bar{K} V)(\bar{K}_l V - \bar{K} V)^T. \quad (8)$$

Similarly, we have

$$\sum_i \|L^T K_i V - L^T \bar{K}_{l_i} V\|_F^2 = \text{Trace}(L^T S_W^V L)$$

where

$$S_W^V = \sum_i (\bar{K}_i V - \bar{K}_{l_i} V)(\bar{K}_i V - \bar{K}_{l_i} V)^T. \quad (9)$$

Then, the optimal L can be computed by solving

$$L^* = \arg\max_{L} \frac{\text{Trace}\left(L^T S_B^V L\right)}{\text{Trace}\left(L^T S_W^V L\right)}. \quad (10)$$

Unlike in KDA, the matrix S_W^V is generally nonsingular here. Commonly, this problem is transformed into the form $\text{Trace}((L^T S_W^V L)^{-1} (L^T S_B^V L))$, and solved with the eigenvalue problem as in traditional LDA [8].

2) V-step: Consider the computation of V for a fixed L. This step is similar to the L-step and we rewrite Criterion 3 as

$$\begin{aligned}V^* &= \arg\max_{V} \frac{\sum_l n_l \|L^T \bar{K}_l V - L^T \bar{K} V\|_F^2}{\sum_i \left\|L^T K_i V - L^T \bar{K}_{l_i} V\right\|_F^2} \\ &= \frac{\text{Trace}\left(V^T S_B^L V\right)}{\text{Trace}\left(V^T S_W^L V\right)}\end{aligned} \quad (11)$$

where

$$\begin{aligned}S_B^L &= \sum_l n_l (\bar{K}_l - \bar{K})^T L L^T (\bar{K}_l - \bar{K}) \\ S_W^L &= \sum_i (K_i - \bar{K}_{l_i})^T L L^T (K_i - \bar{K}_{l_i}).\end{aligned} \quad (12)$$

In addition, similar to the L-step, the optimal solution of V can be calculated as traditional LDA [8].

Those two steps of iterative solution are widely used to solve the optimal problem which has the following form:

$$(L^*, V^*) = \arg\max_{L,V} \frac{\|L^T \times M_{\text{up}} \times V\|_F^2}{\|L^T \times M_{\text{down}} \times V\|_F^2}$$

where M_{up} and M_{down} are two given matrices. In dimensionality reduction, Ye [42] first used this method to solve the problem known as "2-D LDA." To the best of our knowledge, though this iterative solution works well in practice, there is still no theoretical discussion concerning the convergence issue in this context.

The so-called FKA procedure is developed using the iterative optimization method provided previously, and the whole procedure is shown in Fig. 2.

The parameter f' defines the number of the selected combinations of the kernels in the kernel bank. The parameter q actually defines the dimension of projection space as in the traditional KDA algorithm, i.e., the number of discriminant features used for classification. However, the q here is much smaller than the corresponding feature number in KDA, since we have f' selected combinations of kernels and each have q features. Thus, using the projection matrices found by our algorithms, we map each data sample to a $q \times f'$ feature matrix which is used in the classification. Such a feature matrix can be seen to be composed of q feature vector of dimension f', while in KDA, we only have q feature point for each sample. In the real problem of application, we can choose q and f' using crossing validation. Besides, from the result in Section V, we can also see that the algorithm is not sensitive to q and f' within a considerable area.

D. Algorithmic Analysis

Using two projection spaces, our FKC carries out kernel selection and discriminant analysis at the same time. These two projections not only facilitate the computation of the final optimal projection matrix but also provide a unified framework that can be applied to other kernel algorithms such as kernel principle component analysis (KPCA) [24]. Besides, since our algorithm finds all the projection directions at the same time, it is optimal according to our criterion, while in most other algorithms [10], only the first dimension of the projection matrix can be guaranteed to be optimal. On the other hand, our kernel selection is a combination of various different kinds of kernel functions, so it can be regarded as a synthesis of several different classifiers.

Moreover, our proposed solution FKA with FKC could avoid the ill-posed problem in traditional KDA and is easy to implement. Take FKA01 for example; suppose that V^* is the optimal V matrix. Note that $K_iV^* \in \Re^{n\times f}$ and $S_W^{V^*} = \sum_i (K_iV^* - \bar{K}_{l_i}V^*)(K_iV^* - \bar{K}_{l_i}V^*)^T \in \Re^{n\times n}$. Therefore, $\mathrm{rank}(S_W^V)$ is approximately $\min((n-p)*f, n)$. When using multiple kernels, we always have $f > 1$, and hence, often $(n-p)*f \gg n$. Under such a condition, matrix S_W^V is usually of full rank and the problem is not ill-posed.

In addition, the computational complexity of our algorithms is almost the same as the traditional KDA. In FKA01, we need to calculate two kinds of matrix decomposition. In the left step, we compute the decomposition of an $n \times n$ matrix for t_n times, and therefore, the computational complexity is $t_n \times O(n^3)$, while in the right step, the size of right matrix is $f \times f$ and the computational complexity of right step is $t_n \times O(f^3)$. Since mostly $f \ll n$, the whole process of FKA01 is $t_n \times O(n^3)$. As for FKA02, the left step has a decomposition of a $pf \times pf$ matrix for t_n times and the right step of FKA02 is the same with that of FKA01. Therefore, the whole process of FKA02 also has a computational complexity of $t_n \times O((pf)^3)$. In KDA algorithm with a given kernel function, the size of a kernel matrix is $n \times n$, and it needs a time complexity of $O(n^3)$ to find the projection matrix. Hence, to find an optimal kernel function out of f kernel functions for KDA, the time complexity is $f \times O(n^3)$.

In this paper, the number of iterations t_n in our algorithms is set as 100. Though our algorithms may be more time-consuming than selecting a single best kernel from the kernel bank, they can still be done rather quickly in practice. Moreover, since even the number of kernels for a given group of function is usually infinite, to find the optimal kernel out of several predetermined kernels is not a convincing kernel selection method. By contrast, our methods[2] aim to find an optimal combination of different kernels by using the whole kernel bank. From the experiments in Section V, we see that the result of our method is always better than that of using a single optimal kernel in the kernel bank.

V. Experiments

In this section, several experiments are presented to evaluate the performance of the proposed FKA algorithm. We choose the face data to verify the stability of our algorithm. The face databases include the benchmark face databases such as Olivetti and Oracle Research Laboratory (ORL, Cambridge, U.K.) database [25], face recognition technology (FERET) database [27], and Carnegie Mellon University (CMU, Pittsburgh, PA) pose illumination and expression (PIE) database [30]. For simplicity, each experiment is named as Gm/Pn, which denotes that m images per person are randomly selected as the gallery set and other n for the probe set. In our experiment, histogram equilibrium is applied as preprocessing, and then, nearest neighbor is used as the final classifier.

In all the experiments, we first construct the so-called kernel bank, which is a set of various kernel functions. Specifically, we use two different kernel banks: for experiments in Sections V-A–V-C, we adopt Gaussian kernel to construct the Gaussian kernel bank

$$k^i(x,y) = \exp(-\|x-y\|^2/2\sigma_i^2)$$

and set the parameter $\sigma_i = i \times \sigma, i \in \{1,\ldots,10\}$, where σ is the standard deviation of the training data; as for experiments in Section V-D, we employ two kinds of different kernel functions to construct the multikernel bank

$$k^i(x,y) = \begin{cases} \exp(-\|x-y\|^2/2\sigma_i^2), & i = 1,2 \\ (x\cdot y+1)^{(i-2)}, & i = 3,4 \end{cases}$$

where $\sigma_i = i \times \sigma, i = 1,2$. Although decision of the range of σ used in the Gaussian kernel is still an open problem, we may define some reasonable values for σ. In Gaussian kernel, if the value of σ is very small, then $\|x-y\|^2/2\sigma$ is very large, so we have $\exp(-\|x-y\|^2/2\sigma) \to 0$; if the value of σ is very large, then $\|x-y\|^2/2\sigma$ will be very small and $\exp(-\|x-y\|^2/2\sigma) \to 1$. Therefore, we can first identify a lower bound and an upper bound for σ and select σ_i between these two values. The kernel functions with these parameters are used to construct the kernel bank. One may also construct the kernel bank according to some prior knowledge.

In our experiments, we compare the classification result of our FKA with the traditional KDA using the kernel functions in the kernel bank. Though, in general, FKA is unable to decide which kernel to incorporate into the initial kernel bank, as long as the kernel bank has enough kernel functions, the final result of FKA will usually be satisfactory. More importantly, our FKA should perform better than or at least equal to the performance of traditional KDA using only a single kernel from the kernel

[2] If we constrain the right matrix to be a vector with only one nonzero element, our method shall equal to select a single kernel out of the kernel bank. Therefore, we regard our method as a kernel selection method in general.

Fig. 3. Five images of one person in the ORL face database.

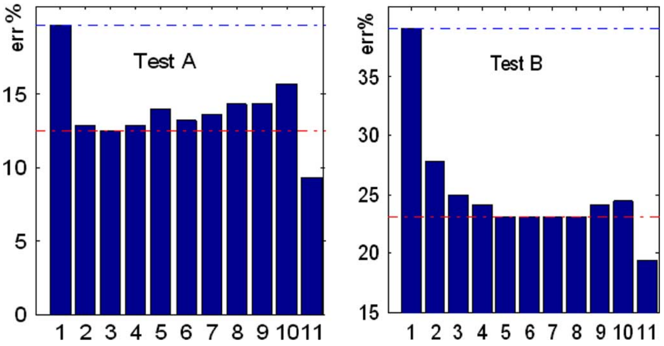


Fig. 4. Error rate of FKA02 versus KDA on the ORL database, using Gaussian kernel bank. The numbers 1–10 represent the result of traditional KDA in the kernel bank and 11 is the result of FKA02.

bank, which shows that the idea of combining various kernels is rather promising in kernel selections.

A. Experiment of Face-Recognition on the ORL Database

The ORL database contains 400 images of 40 individuals. Five sample images of one person in the ORL database are shown in Fig. 3. The images were taken with a tolerance for some tilting and rotation of the faces up to 20°. Some images were captured at different times and had different variations including expression and facial details. All images are grayscale and normalized to a resolution of 46 × 56 pixels.

The whole database is divided into gallery and probe sets as: test A G3/P7, test B G2/P8, and test C G5/P5. Our comparison result between KDA and FKA on the ORL data is presented in Figs. 4 and 5. These two figures display the error rates of the three sets tests: tests A, B, and C. The vertical axis represents error rate of different algorithms. The horizontal axis represents different algorithms, where the numbers from 1 to 10 denote KDA algorithms using ten different parameters while 11 and 12 represent FKA02 and FKA01 with ten kernels in KDA as the kernel bank, respectively. From Fig. 4, we can see that FKA02 has a much lower error rate than the KDAs with different kernels in both cases.

In Fig. 5, we show the results of both FKA01 and FKA02 in test C to compare with ten KDAs with different kernels. It also confirms that both FKA algorithms outperform all the KDAs.

Fig. 6 displays recognition rates of FKA02 on different dimensions and the results are from test C. The horizontal axes represent the row and column dimensions of the final low dimensional matrix used for face recognition.

From Fig. 6, we can see that for a sensible area where q and f' are chosen, the slight change of q and f' will not greatly influence the performance of our FKA algorithm.

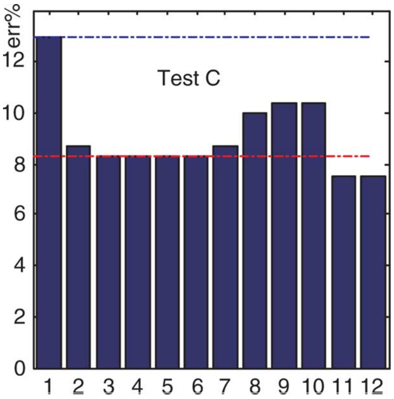


Fig. 5. Error rate of FKA02 and FKA01 versus KDAs on the ORL database, using Gaussian kernel bank. Note that on the horizontal axis, numbers 1–10 represent the result of KDA using individual kernel in the kernel bank, while number 11 represents FKA02 and 12 represents FKA01.

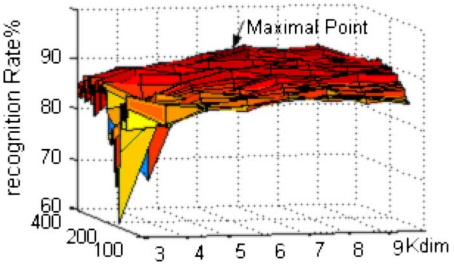


Fig. 6. Accuracy of FKA02 versus number of eigenvectors in test C on the ORL database, using Gaussian kernel bank. The horizontal axes represent the value of the parameter q and f', while the vertical axis represents the recognition rate of FKA02.

Fig. 7. Images of one subject in the FERET database.

B. Experiment of Face Recognition on the FERET Database

The FERET [27] face image database is a standard database for testing and evaluating the state-of-the-art face-recognition algorithms. In our experiment, we use a subset of the FERET database. The subset includes 70 persons and each person has six different facial images. All the images are aligned by fixing the locations of the two eyes and resized to 46 × 56 pixels. Facial expressions, illumination, pose, and facial details vary in the images. Fig. 7 displays six examples of one person in FERET.

We randomly partition the database into G4/P2 as test A and G2/P4 as test B. Fig. 8 presents the results of these two experiments, which also demonstrates that the algorithm FKA02 outperforms the traditional KDA algorithms with different kernel parameters.

C. Experiment of Face Recognition on the PIE Database

The CMU PIE database contains more than 40 000 facial images of 68 people. The images were acquired with different poses, under various illumination conditions, and with different facial expressions.

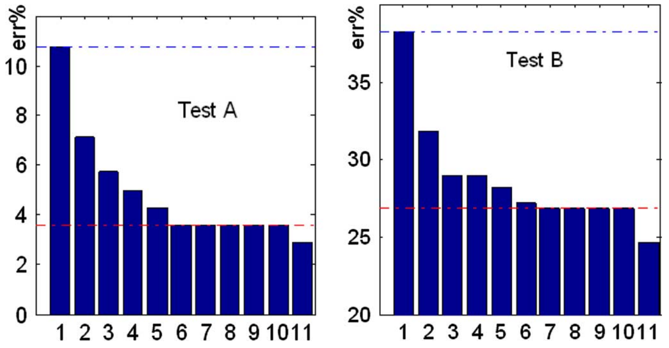


Fig. 8. Error rate of FKA02 versus KDA on the FERET database, using Gaussian kernel bank. The numbers 1–10 represent the result of KDA using individual kernel in the kernel bank, while the number 11 represents the result of FKA02.

Fig. 9. Five images of one person in the PIE database.

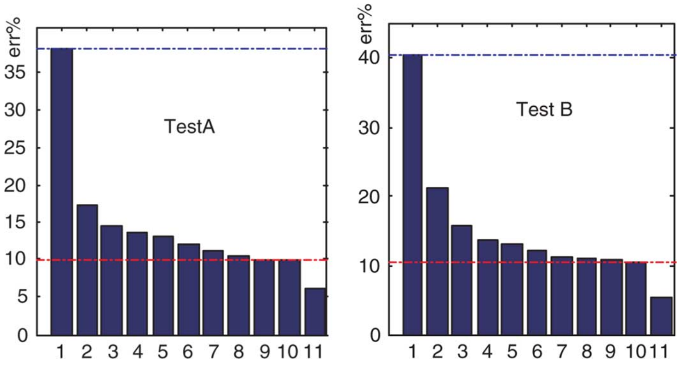


Fig. 10. Error rate of FKA01 versus KDA on the PIE database, using Gaussian kernel bank. The numbers 1–10 represent the result of KDA using individual kernel in the kernel bank, while the number 11 represents the result of FKA01.

Fig. 9 shows five images of one person without preprocessing. In our used database of PIE, five near frontal poses (C27, C05, C29, C09, and C07) and illumination 08 and 11 are chosen. The flash 08 and 11 are placed near the center and the illumination can be considered as the nearly frontal illumination. Each person has ten images and all the images are aligned by fixing the locations of two eyes, and the images are resized to 64 × 64 pixels.

Similar to the previous experiments, the data set is randomly partitioned into gallery and probe sets with G4/P6 in test A and G3/P7 in test B. We compare FKA01 with KDA in this experiment.

The result in Fig. 10 again shows that FKA01 improves face-recognition accuracy compared to traditional KDA with different kernel parameters.

D. *Experiment of Multikernel Functions on the ORL and the FERET Database*

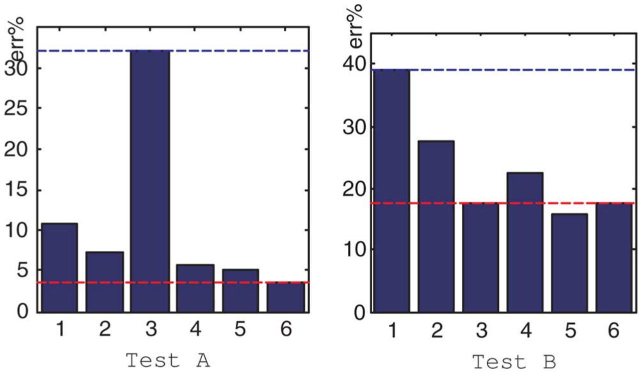


Fig. 11. Error rate of FKA01 and FKA02 versus KDAs using the multikernel bank. The numbers 1–4 represent the results of KDAs using individual kernel functions in multikernel bank. The number 5 represents the result of FKA01 and 6 represents the result of FKA02.

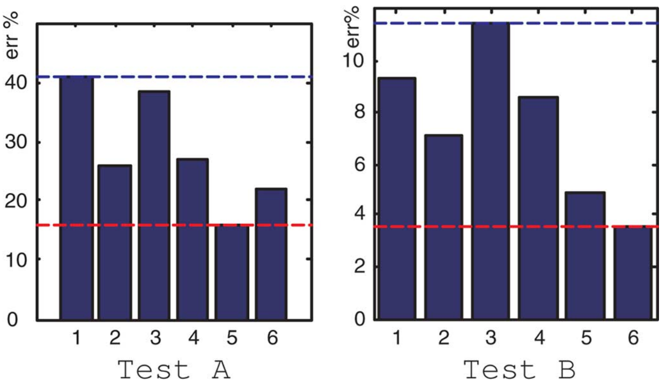


Fig. 12. Error rate of FKA01 and FKA02 versus KDAs and RKDA using the multikernel bank. The numbers 1 and 2 represent the results of KDAs using Gaussian kernel functions, the numbers 3 and 4 represent the result of RKDA using the same Gaussian kernel, and the numbers 5 and 6 represent the result of FKA01, 02 using the whole multikernel bank.

In this section, we use two kinds of kernel functions in our multikernel bank in order to justify that our FKA is an effective kernel selection method among different kernels. We present two sets of experiments using the subset of ORL, which is partitioned as G2/P8, and a subset of FERET database, which is partitioned as G4/P2.

Fig. 11 is the results of experiment on the FERET and ORL database, showing the comparison between the results of FKA and KDA. Test A is the result on the G4/P2 subset of the FERET database, while test B is the result on the G2/P8 subset of the ORL database. From Fig. 11, we can see that the results of our FKA are better than those of the KDAs.

We also compare our FKA algorithm with the method in [16]. In Fig. 12, we compare our FKA algorithm with KDA and RKDA using the kernel in the multikernel bank on the PIE database. Test A is the result on the G2/P8 subset of the ORL database and test B is the result on G4/P2 FERET.

From all the previous experiments, we find that the parameter for KDA to obtain the best performance is different on different data set, and the same kernel function may have different performance on different data sets. Hence, it is difficult and unreasonable to select kernels empirically as in traditional KDA. While our algorithms can effectively combine different kernels and derive elegant representation for classification, thus are superior to the traditional KDA in almost all cases.

For KDA, there are some methods [6], [12] that process kernel selection under the framework of adding a penalized term for regularization. Although these methods can be solved efficiently by using convex optimization, there is still problem on how to optimize the regularization term and kernel function simultaneously. Rather than using penalized term, we directly restrict the projection space to avoid overfitting, which can also be viewed as a kind of regularization in general. By using that restriction, we can successfully avoid overfitting and at the same time provide a feasible framework for kernel selection.

VI. Conclusion and Discussion

In this paper, we have proposed a novel FKC to fuse kernel selection into the process of discriminant analysis. This criterion automatically combines different types of kernels to search for the most discriminating features. We also propose two iterative algorithms called FKA01 and FKA02 to optimize the new criterion. In the two algorithms, we use the constrained solution rather than the traditional regularization methods to solve the ill-posed problem. On one hand, the constrained solution reduces the overhead of computation; on the other hand, although the constrained solution cannot solve the ill-posed problem in theory, it alleviates the ill-posed problem to some extent. Extensive experiments on different databases show the superiority of the proposed FKA.

One shortcoming of FKA is that it is iterative and may suffer from the local minima. Future research projects could study how to use the stimulated annealing method [2] to search for the global optimum. In addition, the methods can also be combined with other linear subspace methods [35], [36].

Appendix I
Scatter Matrix

Denote $\Phi = (\phi(x_1), \ldots, \phi(x_n))$ and the scatter matrices given in (2) can be rewritten as

$$\begin{aligned} S_W = \Phi & \left[\sum_{i=1}^{n} \left(e_i - \frac{1}{n_{l_i}} \sum_{j=1}^{n} \delta(l_i, l_j) e_j \right) \right. \\ & \qquad \left. \cdot \left(e_i - \frac{1}{n_{l_i}} \sum_{j=1}^{n} \delta(l_i, l_j) e_j \right)^T \right] \Phi^T \\ \triangleq & \, \Phi M_W \Phi^T \end{aligned} \tag{13}$$

where

$$\delta(i, j) = \begin{cases} 1, & \text{if} \quad i{=}j \\ 0, & \text{else} \end{cases}$$

and e_i is an n-dimensional vector with $e_i(j) = \delta_{ij}$. Let S_T be the total scatter matrix as

$$\begin{aligned} S_T = \Phi & \sum_{i=1}^{n} \left(e_i - \frac{1}{n} \sum_{j=1}^{n} e_j \right) \left(e_i - \frac{1}{n} \sum_{j=1}^{n} e_j \right)^T \Phi^T \\ \triangleq & \, \Phi M_T \Phi^T. \end{aligned}$$

Then, S_B can be written as

$$S_B = S_T - S_W = \Phi (M_T - M_W) \Phi^T \triangleq \Phi M_B \Phi^T. \tag{14}$$

Appendix II
Proof of Overfitting

Lemma 1: The rank of matrix M_W is $n - p$.

Example: In KDA algorithms, when K is a nonsingular matrix, there exist p basis vectors that map the original data of the same class to the same point.

Solution: When K is nonsingular, (4) can be simplified as $M_B K \alpha = \lambda M_W K \alpha$. Set that $K\alpha = \beta, \beta \in \Re^n$. Using Lemma 1, we conclude that M_W has only p zero eigenvalues.

Define

$$\beta_l = (\overbrace{0, \ldots, 0}^{\sum_{j=1}^{l-1} n_j}, \overbrace{1, \ldots, 1}^{n_l}, 0, \ldots, 0)$$

and we have

$$M_w \beta_l = \mathrm{diag}(0, \ldots, 0, A_l, B_l, 0, \ldots, 0) = 0. \tag{15}$$

This means that β_l $(l = 1, \ldots, p)$ are the eigenvectors corresponding to M_W's zero eigenvalues. Recalling the M matrices defined in (13) and (14), we can obtain

$$M_B \beta_l = (M_T - M_W) \beta_l = M_T \beta_l \neq 0. \tag{16}$$

Choose $\alpha_l = K^{-1} \beta_l$ as the eigenvectors, and we have $\lambda = \infty$.

The image of x_i in higher dimensional feature space $\phi(x_i)$ is projected to $g_l(x_i) = \phi(x_i)\psi_l$. Note that

$$g_l(x_i) = \phi(x_i) \sum_{j=1}^{n} \alpha_l(j) \phi_j = K_i \cdot \alpha_l = \beta_l(i)$$

where $\beta_l(i)$ denotes the ith entry of the vector β_l. Let $\psi_l = \sum_{j=1}^{n} \alpha_l(j) \phi_j$ be a set of basis vectors of the feature space, where the vector α_l corresponds to $K\alpha_l = \beta_l$. Note that when x_i, x_j are two different samples from the same class (for example, class c) in the original data set, they are projected to the coordinate 0 under the basis $\beta_l (l \neq c)$ and the coordinate 1 under the basis $\beta_i (i = c)$. Under this set of the basis, the data of the same class is mapped to the same data point in the reduced-dimension subspace.

Appendix III
Proof of Reproducing Hilbert Space

Define the addition of two vectors

$$\hat{\phi}(x) = [\phi_1(x)^T, \ldots, \phi_f(x)^T]^T$$

and

$$\hat{\phi}(y) = [\phi_1(y)^T, \ldots, \phi_f(y)^T]^T$$

in $\mathcal{H}$ as

$$\hat{\phi}(x) \oplus \hat{\phi}(y) {=} [(\phi_1(x){+}\phi_1(y))^T, \ldots, (\phi_f(x){+}\phi_f(y))^T]^T$$

and the scalar multiplication between an element $\hat{\phi}(x) \in \mathcal{H}$ and a scalar λ as

$$\lambda \otimes \hat{\phi}(x) = [\lambda\phi_1(x)^T, \ldots, \lambda\phi_f(x)^T].$$

The inner product between $\hat{\phi}(x)$ and $\hat{\phi}(y)$ is given as

$$\langle \hat{\phi}(x), \hat{\phi}(y) \rangle = \sum_{l=1}^{f} \phi_l(x)^T \phi_l(y).$$

First, we prove that $(\mathcal{H}, \langle\cdot\rangle)$ is a Hilbert space by showing that the induced metric $||\cdot||$ is complete in $\mathcal{H}$. Now, suppose that $\hat{\phi}(x^i)$ is a sequence of elements in $\mathcal{H}$ which satisfies

$$\lim_{n,m\to\infty} ||\hat{\phi}(x^n) - \hat{\phi}(x^m)||^2 = 0.$$

Since

$$||\hat{\phi}(x^n) - \hat{\phi}(x^m)||^2 = \sum_{l=1}^{f} ||\phi_l(x^n) - \phi_l(x^m)\ ||^2$$

we have

$$\lim_{n,m\to\infty} ||\phi_l(x^n) - \phi_l(x^m)||^2 = 0, \qquad \text{for each } \ l = 1, \ldots, f.$$

Recall that each $\mathcal{H}_i$ is a Hilbert space, thus we know that there exists an element $\phi_i(x)$ in $\mathcal{H}_i$ satisfying

$$\lim_{n\to\infty} ||\phi_i(x^n) - \phi_i(x)\ ||^2 = 0.$$

Now, let $\hat{\phi}(x) = [\phi_1(x)', \ldots, \phi_f(x)']'$ be an element in $\mathcal{H}$. From previous deduction, we immediately know

$$\lim_{n\to\infty} ||\hat{\phi}(x^n) - \hat{\phi}(x)||^2 = \sum_{i=1}^{f} \lim_{n\to\infty} ||\phi_i(x^n) - \phi_i(x)||^2 = 0$$

which proves that $(\mathcal{H}, \langle\cdot\rangle)$ is a Hilbert space.

Denote $k(\cdot, x) = \sum_{i=1}^{f} k_i(\cdot, x)$, where each k_i is the kernel function corresponding to $\phi_i(x)$, now enough to justify the reproducing property of $\mathcal{H}, \langle\cdot\rangle$. Notice that $f \in \mathcal{H}$ and we can denote $f(\cdot) = \sum_{j=1}^{m} \alpha_j k(\cdot, x_j)$, where $\alpha_j \in \Re$ and $x_1, \ldots, x_m \in \chi$ are arbitrary. We have

$$\begin{aligned}
\langle f, k(x,\cdot)\rangle &= \left\langle \sum_{j=1}^{m} \alpha_j k(\cdot, x_j), k(x,\cdot) \right\rangle \\
&= \sum_{j=1}^{m} \alpha_j \langle k(\cdot, x_j), k(x,\cdot)\rangle \\
&= \sum_{j=1}^{m} \alpha_j \left\langle \sum_{r=1}^{f} k_r(\cdot, x_j), \sum_{s=1}^{f} k_s(x,\cdot) \right\rangle \\
&= \sum_{j=1}^{m} \alpha_j \left[\sum_{r=1}^{f} \left(\sum_{s=1}^{f} \langle k_r(\cdot, x_j), k_s(x,\cdot) \langle \right) \right] \\
&= \sum_{j=1}^{m} \alpha_j \sum_{r=1}^{f} k_r(x, x_j) = \sum_{j=1}^{m} \alpha_j k(x, x_j) = f(x)
\end{aligned}$$

which proves that $(\mathcal{H}, \langle\cdot\rangle)$ is a reproducing kernel Hilbert space.

References

[1] G. Baudat and F. Anouar, "Generalized discriminant analysis using a kernel approach," *Neural Comput.*, vol. 12, no. 10, pp. 2385–2404, 2000.

[2] G. Casella and C. Robert, *Monte Carlo Statistical Methods*. New York: Springer-Verlag, 1999.

[3] K. Crammer, J. Keshet, and Y. Singer, "Kernel design using boosting," in*Advances in Neural Information Processing Systems*, ser. 15. Cambridge, MA: MIT Press.

[4] N. Cristianini, J. Kandola, A. Elisseeff, and J. Shawe-Taylor, "On kernel-target alignment," in *Proc. Neural Inf. Process. Syst.*, Dec. 2001, pp. 367–373.

[5] T. Evgeniou, M. Pontil, and A. Elisseeff, "Leave one out error, stability, and generalization of voting combinations of classifiers," *Mach. Learn.*, vol. 55, no. 1, pp. 71–97, Apr. 2004.

[6] G. Fung, M. Dundar, J. Bi, and B. Rao, "A fast iterative algorithm for Fisher discriminant using Heterogeneous kernels," in *Proc. 21st Int. Conf. Mach. Learn.*, Banff, Canada, 2004, pp. 313–320.

[7] T. Gestel, J. Suyken, G. Lanckriet, A. Lambrechts, B. De Moor, and J. Vanderwalle, "Bayesian framework for least squares support vector machine classifiers, Gaussian process and kernel Fisher discriminant analysis," *Neural Comput.*, vol. 15, no. 5, pp. 1115–1148, May 2002.

[8] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. New York: Springer-Verlag, 2001.

[9] T. Hastie and R. Tibshirani, "Discriminant analysis by Gaussian mixture," *J. Roy. Statist. Soc.*, Jan. 1996.

[10] H. Jiang, C. Pong, W. Chen, and J. Lai, "Kernel subspace LDA with optimized kernel parameters on face recognition," in *Proc. 6th IEEE Int. Conf. Autom. Face Gesture Recognit.*, 2004, pp. 327–332.

[11] R. Kondor and T. Jebara, "A kernel between sets of vectors," in *Proc. 20th Int. Conf. Mach. Learn.*, Washington, DC, 2003, pp. 361–368.

[12] S. Kim, A. Magnani, and S. Boyd, "Optimal kernel selection in kernel Fisher discriminant analysis," in *Proc. Int. Conf. Mach. Learn.*, Pittsburgh, PA, 2006, pp. 465–472.

[13] M. Kirby and L. Sirovih, "Application of the Karhunen–Loeve procedure for the characterization of human faces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 1, pp. 103–108, Jan. 1990.

[14] W. J. Krzanowski, P. Jonathan, W. V. McCarthy, and M. R. Thomas, "Discriminant analysis with singular covariance matrices: Methods and applications to spectroscopic data," *Appl. Statist.*, no. 44, pp. 887–894, 2004.

[15] G. Lanckriet, N. Cristianini, L. Ghaoui, P. Bartlett, and M. Jordan, "Learning the kernel matrix with semi-definite programming," *J. Mach. Learn. Res.*, vol. 5, pp. 27–72, 2004.

[16] Z. Liang, D. Zhang, and P. Shi, "Robust kernel discriminant analysis and its application to feature extraction and recognition," *Neurocomput.*, vol. 69, no. 7–9, pp. 928–933, 2006.

[17] Q. Liu, H. Lu, and S. Ma, "Improving kernel Fisher discriminant analysis for face recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 1, pp. 42–49, Jan. 2004.

[18] Q. Liu, X. Tang, H. Lu, and S. Ma, "Face recognition using kernel scatter-difference-based discriminant analysis," *IEEE Trans. Neural Netw.*, vol. 17, no. 4, pp. 1081–1085, Jul. 2006.

[19] J. Lu, K. Plataniotis, and N. Venetsanopoulos, "Face recognition using kernel direct discriminant analysis algorithms," *IEEE Trans. Neural Netw.*, vol. 14, no. 1, pp. 117–126, Jan. 2003.

[20] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K. R. Muller, "Fisher discriminant analysis with kernels," in *Proc. IEEE Int. Workshop Neural Netw. Signal Process. IX*, Aug. 1999, pp. 41–48.

[21] S. Mika, G. Ratsch, B. Scholkopf, A. Smola, J. Weston, and K. R. Muller, "Invariant feature extraction and classification in kernel spaces," in *Advances in Neural Information Processing Systems*, ser. 12. Cambridge, MA: MIT Press, 1999.

[22] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, A. Smola, and K. R. Muller, "Constructing descriptive and discriminative nonlinear features: Rayleigh coefficients in kernel feature space," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 623–628, May 2003.

[23] S. Mika, G. Ratsch, and K. R. Muller, "A mathematical programming approach to the kernel Fisher algorithm," in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 2001, vol. 13, pp. 591–597.

[24] K. R. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkof, “An introduce to kernel based learning algorithm,” *IEEE Trans. Neural Netw.*, vol. 12, no. 2, pp. 181–201, Mar. 2001.

[25] Olivetti and Oracle Research Laboratory, “The Olivetti and Oracle Research Laboratory face database of faces,” Cambridge, U.K. [Online]. Available: http://www.cam-orl.co.uk/facedatabase.html

[26] J. Peltonen and S. Kaski, “Discriminative components of data,” *IEEE Trans. Neural Netw.*, vol. 16, no. 1, pp. 68–83, Jan. 2005.

[27] P. J. Philips, H. Wechsler, J. Huang, and P. Rauss, “The FERET database and evaluation procedure for face recognition algorithms,” *Image Vis. Comput.*, vol. 16, pp. 295–306, 1998.

[28] V. Roth and V. Steinhage, “Nonlinear discriminant analysis using kernel functions,” in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 2000, vol. 12, pp. 568–574.

[29] B. Scholkopf and A. Smola, *Learning With Kernels*. Cambridge, MA: MIT Press, 2002.

[30] T. Sim, S. Baker, and M. Bsat, “The CMU pose, illumination, and expression (PIE) database,” in *Proc. IEEE Int. Conf. Autom. Face Gesture Recognit.*, May 2002, pp. 53–58.

[31] J. Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[32] A. N. Tikhonov and V. Y. Arsenin, *Solution of Ill-Posed Problems*. New York: Wiley, 1997.

[33] V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.

[34] V. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.

[35] X. Wang and X. Tang, “A unified framework for subspace face recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1222–1228, Sep. 2004.

[36] X. Wang and X. Tang, “Random sampling for subspace face recognition,” *Int. J. Comput. Vis.*, vol. 70, no. 1, pp. 91–104, Oct. 2006.

[37] A. R. Webb, *Statistical Pattern Recognition*, 2nd ed. New York: Wiley, 2004.

[38] H. Xiong, M. N. S. Swamy, and M. O. Ahmad, “Optimizing the kernel in the empirical feature space,” *IEEE Trans. Neural Netw.*, vol. 16, no. 2, pp. 460–474, Mar. 2005.

[39] M. H. Yang, “Kernel eigenfaces vs. kernel Fisherfaces: Face recognition using kernel methods,” in *Proc. 5th Int. Conf. Autom. Face Gesture Recognit.*, Washington, D.C., May 2002, pp. 215–220.

[40] J. Yang, A. F. Frangi, J. Yang, D. Zhang, and Z. Jin, “KPCA plus LDA: A complete kernel Fisher discriminant framework for feature extraction and recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 2, pp. 230–244, Feb. 2005.

[41] J. Ye, R. Jananrdan, C. H. Park, and H. Park, “An optimization criterion for generalized discriminant analysis on undersampled problems,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 8, pp. 982–994, Aug. 2004.

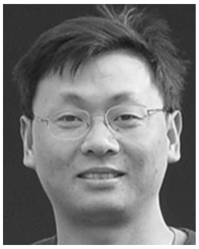
[42] J. Ye, R. Janardan, and Q. Li, “Two-dimensional linear discriminant analysis,” in *Proc. Neural Information Processing Systems*, 2005, pp. 1596–1576.

[43] W. Zhao, R. Chellappa, and P. J. Phillips, “Subspace linear discriminant analysis for face recognition,” Ctr. Autom. Res., Univ. Maryland, College Park, MD, Tech. Rep., 1999.

[44] W. Zheng, L. Zhao, and C. Zou, “Foley-Sammon optimal discriminant vectors using kernel approach,” *IEEE Trans. Neural Netw.*, vol. 16, no. 1, pp. 1–9, Jan. 2005.

Shu Yang received the M.S. degree from the School of Mathematical Science and the B.S. degree from the Department of Artificial Intelligence, Peking University, Beijing, China, in 2004 and 2006, respectively. Currently, she is working towards the Ph.D. degree at the Department of Mathematics and Statistics, Boston University, Boston, MA.

Her major interests include statistical learning and biostatistics.



Shuicheng Yan (M’06) received the B.S. and Ph.D. degrees from the Applied Mathematics Department, School of Mathematical Sciences, Peking University, Beijing, China, in 1999 and 2004, respectively.

His research interests include computer vision and machine learning.

Chao Zhang received the B.Eng., M.S., and Ph.D. degrees in electrical engineering from the Northern Jiaotong University, Beijing, China, in 1984, 1989, and 1995, respectively.

After working as a Postdoctoral Research Fellow for two years, he became a faculty member in June 1997 at the National Laboratory on Machine Perception, Peking University, Beijing, China, where he is currently an Associate Professor. His research interests include computer vision, statistical pattern recognition, and video-based biometrics.

Xiaoou Tang (S’93–M’96–SM’02) received the B.S. degree from the University of Science and Technology of China, Hefei, Chinam in 1990, the M.S. degree from the University of Rochester, Rochester, NY, in 1991, and the Ph.D. degree from the Department of Computer Science and Engineering, Massachusetts Institute of Technology, Cambridge, in 1996.

He is a Professor at the Department of Information Engineering, the Chinese University of Hong Kong, Hong Kong, and the Group Manager of the Visual Computing Group at the Microsoft Research Asia, Beijing, China. His research interests include computer vision, pattern recognition, and video processing.

Dr. Tang is a local chair of the IEEE International Conference on Computer Vision (ICCV 2005), an area chair of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2007), a program chair of ICCV 2009, and a general chair of the IEEE ICCV International Workshop on Analysis and Modeling of Faces and Gestures 2005. He has been a Guest Editor for the IEEE JOURNAL OF OCEANIC ENGINEERING and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY. He is an Associate Editor of the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and *Pattern Recognition Journal*.