Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 898560, 9 pages
http://dx.doi.org/10.1155/2014/898560

Research Article

Multiple Data-Dependent Kernel Fisher Discriminant Analysis for Face Recognition

Yue Liu,[1] Yibing Li,[1] Hong Xie,[1] and Dandan Liu[2]

[1] *School of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China*
[2] *Institute of Electrical and Control Engineering, Heilongjiang University of Science and Technology, Harbin 150001, China*

Correspondence should be addressed to Yibing Li; liyibing0920@sina.cn

Received 15 May 2014; Accepted 16 July 2014; Published 6 August 2014

Academic Editor: Mohamed A. Seddeek

Kernel Fisher discriminant analysis (KFDA) method has demonstrated its success in extracting facial features for face recognition. Compared to linear techniques, it can better describe the complex and nonlinear variations of face images. However, a single kernel is not always suitable for the applications of face recognition which contain data from multiple, heterogeneous sources, such as face images under huge variations of pose, illumination, and facial expression. To improve the performance of KFDA in face recognition, a novel algorithm named multiple data-dependent kernel Fisher discriminant analysis (MDKFDA) is proposed in this paper. The constructed multiple data-dependent kernel (MDK) is a combination of several base kernels with a data-dependent kernel constraint on their weights. By solving the optimization equation based on Fisher criterion and maximizing the margin criterion, the parameter optimization of data-dependent kernel and multiple base kernels is achieved. Experimental results on the three face databases validate the effectiveness of the proposed algorithm.

1. Introduction

Face recognition has received extensive attention in many image processing applications. In these applications, the original face images commonly lie in a high-dimension space, resulting in low recognition accuracy and high cost. Existing image feature extraction algorithms can roughly fall into two categories: feature extraction based on signal processing and learning-based feature extraction [1–4]. By the utilization of learning-based approach, the original images can be mapped into a lower-dimensional feature space in which the essential structure of the original space becomes clear. To this end, Fisher discriminant analysis (FDA) [5], principal component analysis (PCA), and locality preserving projection (LPP) [6] are typical learning-based feature extraction techniques. Moreover, one of the most famous algorithms applied in face recognition is Fisher face, which is based on a two-phase framework: PCA plus LDA [3, 4]. It maximizes the between-class scatter and minimizes the within-class scatter to separate one class from the others. However, the entire above mentioned algorithms are linear subspaces analysis methods in essence, so they are inadequate to depict the complexity face images. To overcome the limitation, many nonlinear algorithms, such as kernel-based PCA (KPCA) [7] and FDA (KFDA) [8], have been devised and attained good performance in face recognition. It has been demonstrated that KFDA is a feasible nonlinear feature extraction algorithm for face recognition. However, the performance of KFDA is sensitive to the kernel function selection and its parameters. Moreover, the ability of single kernel is quite limited in depicting geometrical structure of some aspects for the input data. Once the face images are captured under huge variations of pose, illumination, facial expression, and so forth, single kernel-based FDA could not be suitable for the face recognition. In summary, kernel functions play an important part in face recognition applications [9, 10].

As a consequence, various approaches have been developed to handle the above issues, and two main categories are identified as follows. (1) Devise multiple kernels by convex combination of multiple basic kernels. If so, different data descriptors can be used to depict the geometrical structures of original data from multiple views, which can complement to improve recognition performance [11–14]. (2) Develop data-dependent kernel (DK) by conformal transformation of the basic kernel. If so, the designed kernel would be adaptive to the input data, leading to a substantial improvement in the performance of KFDA algorithm [15–17].

In this paper, to improve the performance of KFDA, we proposed a novel feature extraction algorithm for face recognition called multiple data-dependent kernel Fisher discriminant analysis (MDKFDA) based on the multiple kernel learning (MKL). The main contributions of this paper lie in the following. (1) By introducing MDK into KFDA, maximum discrimination performance can be achieved in feature space. (2) Multiple image features extracted in different descriptors are fully utilized in the MDKFDA algorithm. (3) Nonlinear discriminant features are produced by the adoption of MKL.

The rest of this paper is organized as follows. Section 2 shows a brief overview of MKL and KFDA. In Section 3, we illustrate the proposed MDKFDA algorithm and introduce the parameter optimization scheme for data-dependent kernel and multiple base kernels. Extensive experimental results on face recognition are reported in Section 4. Finally, Section 5 concludes this paper.

2. Related Work

In this section, we will briefly introduce some previous works related to this paper, including KFDA and MKL.

2.1. Kernel Fisher Discriminant Analysis. KFDA is a nonlinear feature extraction algorithm which combines the nonlinear kernel trick with FDA. Because of its ability to extract the discriminatory nonlinear features, KFDA and its variations are frequently used for face recognition. In this paper, a two-phased KFDA framework proposed by Yang et al. [18] is adopted to construct the MDKFDA. The two-phased KFDA framework mainly contains two parts: KPCA is applied to reduce the dimension of input space, and then LDA is used to further extract the features in the KPCA-transformed space.

Given the input training sample set including N samples:

$$\mathbf{S} = \left\{(\mathbf{x}_i, c_i) \mid \mathbf{x}_i \in \mathbb{R}^d, c_i \in \{1,\ldots,C\}\right\}_{i=1}^{N}, \tag{1}$$

where $\mathbf{x}_i$ is the training sample of d-dimension and c_i represents the class label of $\mathbf{x}_i$. Given a sample $\mathbf{x}$, its nonlinear mapped image can be denoted as $\Phi(\mathbf{x})$, and the discriminate feature vector $\mathbf{z}$ can be obtained as follows:

$$\mathbf{z} = \mathbf{G}^{\mathrm{T}}\mathbf{P}^{\mathrm{T}}\Phi(\mathbf{x}). \tag{2}$$

Equation (2) contains two transformations $\mathbf{P}$ and $\mathbf{G}$. The transformation $\mathbf{P}$ represents KPCA which transforms the input space $\mathbb{R}^d$ into feature space $\mathbb{R}^p$, while the transformation $\mathbf{G}$ is the Fisher discriminant transformation in the KPCA-transformed space $\mathbb{R}^p$.

Firstly, the issues in the process of KPCA are described as follows.

For a given nonlinear mapping Φ, the input space $\mathbb{R}^p$ can be projected into the feature space $\mathbb{F}$, which is considered as Hilbert space. The covariance operator on the feature space $\mathbb{F}$ can be represented as

$$\mathbf{H} = \frac{1}{N}\sum_{i=1}^{N}\left(\Phi(\mathbf{x}_i) - \overline{\Phi}\right)\left(\Phi(\mathbf{x}_i) - \overline{\Phi}\right)^{\mathrm{T}}, \tag{3}$$

where $\overline{\Phi} = (1/N)\sum_{i=1}^{N}\Phi(\mathbf{x}_i)$. The way to find the nonzero eigenpair of H is illustrated as follows. Previously, to simplify the deduced process, the covariance operator is reformulated as

$$\widetilde{\mathbf{H}} = \frac{1}{N}\sum_{i=1}^{N}\Phi(\mathbf{x}_i)\Phi(\mathbf{x}_i)^{\mathrm{T}}. \tag{4}$$

Let us denote $\mathbf{Q} = [\Phi(\mathbf{x}_1),\ldots,\Phi(\mathbf{x}_N)]$ and construct a $N\times N$ Gram matrix $\widetilde{\mathbf{K}} = \mathbf{Q}^{\mathrm{T}}\mathbf{Q}$, whose element $\widetilde{\mathbf{K}}_{ij}$ can be calculated through the use of kernel tricks:

$$\widetilde{\mathbf{K}}_{ij} = \Phi(\mathbf{x}_i)^{\mathrm{T}}\Phi(\mathbf{x}_j) = \left\langle\Phi(\mathbf{x}_i),\Phi(\mathbf{x}_j)\right\rangle = k(\mathbf{x}_i,\mathbf{x}_j). \tag{5}$$

Centralize $\widetilde{\mathbf{K}}$ by $\mathbf{K} = \widetilde{\mathbf{K}} - \mathbf{1}_N\widetilde{\mathbf{K}} - \widetilde{\mathbf{K}}\mathbf{1}_N + \mathbf{1}_N\widetilde{\mathbf{K}}\mathbf{1}_N$, where $\mathbf{1}_N = (1/N)_{N\times N}$.

We adopt the t largest positive eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_t$ of $\mathbf{K}$ and their corresponding orthonormal eigenvectors $\mathbf{u}_1,\mathbf{u}_2,\ldots,\mathbf{u}_t$ to calculate the eigenvectors $\mathbf{w}_1,\mathbf{w}_2,\ldots,\mathbf{w}_t$ of $\widetilde{\mathbf{H}}$ as follows:

$$\mathbf{w}_j = \frac{1}{\sqrt{\lambda_j}}\mathbf{Q}\mathbf{u}_j, \quad j = 1,2,\ldots,t. \tag{6}$$

Hence, we can get the KPCA-transformed feature vector $\mathbf{y} = (\mathbf{y}_1,\ldots,\mathbf{y}_t)^{\mathrm{T}}$, and the ith KPCA feature is obtained by

$$\begin{aligned}\mathbf{y}_j &= \mathbf{w}_j^{\mathrm{T}}\Phi(\mathbf{x}) \\ &= \frac{1}{\sqrt{\lambda_j}}\mathbf{u}_j^{\mathrm{T}}\mathbf{Q}^{\mathrm{T}}\Phi(\mathbf{x}) \\ &= \frac{1}{\sqrt{\lambda_j}}\mathbf{u}_j^{\mathrm{T}}\left[k(\mathbf{x}_1,\mathbf{x}),k(\mathbf{x}_2,\mathbf{x}),\ldots,k(\mathbf{x}_N,\mathbf{x})\right].\end{aligned} \tag{7}$$

Above all, we can describe the $\mathbf{P}$ as follows:

$$\mathbf{P} = (\mathbf{w}_1,\ldots,\mathbf{w}_t). \tag{8}$$

Secondly, the issues in the process of Fisher discriminant transformation are illustrated as follows.

FDA is used for further feature extraction in the KPCA-transformed space $\mathbb{R}^p$. In order to maximize the Fisher criterion, we first define the between-class scatter operator $\mathbf{S}_b^{\Phi}$ and the within-class scatter operator $\mathbf{S}_w^{\Phi}$ in feature space $\mathbb{F}$. Consider

$$\mathbf{S}_b^{\Phi} = \frac{1}{N}\sum_{i=1}^{c}l_i\left(\mathbf{m}_i^{\Phi} - \mathbf{m}_0^{\Phi}\right)\left(\mathbf{m}_i^{\Phi} - \mathbf{m}_0^{\Phi}\right)^{\mathrm{T}}, \tag{9}$$

$$\mathbf{S}_w^{\Phi} = \frac{1}{N}\sum_{i=1}^{c}\sum_{j=1}^{l_i}\left(\Phi(\mathbf{x}_{ij}) - \mathbf{m}_i^{\Phi}\right)\left(\Phi(\mathbf{x}_{ij}) - \mathbf{m}_i^{\Phi}\right)^{\mathrm{T}}, \tag{10}$$

where l_i is the number of training samples in class i, $\mathbf{x}_{ij}$ represents the jth samples in class i, m_i^Φ is the mean of the mapped samples in class i, and m_0^Φ is the mean across all mapped samples. Thus, we can obtain the Fisher criterion by

$$\mathbf{J}^\Phi(\mathbf{V}) = \frac{\mathbf{V}^{\mathrm{T}}\mathbf{S}_b^\Phi \mathbf{V}}{\mathbf{V}^{\mathrm{T}}\mathbf{S}_w^\Phi \mathbf{V}}, \tag{11}$$

where $\mathbf{V}$ is the discriminant vector. According to Mercer kernel function theory, each $\mathbf{V}$ can be described by the elements of feature space $\{\Phi(\mathbf{x}_1), \Phi(\mathbf{x}_2), \ldots, \Phi(\mathbf{x}_N)\}$, and there always exist coefficients φ_i, $i = 1, 2, \ldots, N$, such that

$$\mathbf{V} = \sum_{i=1}^{N} \varphi_i \Phi(\mathbf{x}_i) = \mathbf{\Psi X}, \tag{12}$$

where $\mathbf{\Psi} = [\Phi(\mathbf{x}_1), \Phi(\mathbf{x}_2), \ldots, \Phi(\mathbf{x}_N)]$, $\mathbf{X} = [\varphi_1, \ldots, \varphi_N]^{\mathrm{T}}$. Hence, the Fisher optimal discriminant vectors are the stationary points $\mathbf{V}_1, \ldots, \mathbf{V}_d$, $d \le N$, and, correspondingly, the transformation $\mathbf{G}$ in (2) can be denoted by

$$\mathbf{G} = (\mathbf{V}_1, \ldots, \mathbf{V}_d). \tag{13}$$

2.2. Multiple Kernel Learning. In general, MKL refers to the process of learning a kernel machine which is the combination of multiple base kernel functions/matrices. Recent research efforts have shown that MKL is not only able to find an optimal combination weight of base kernels but also improve the performance of the resulting classifiers.

As mentioned above, $\mathbf{S}$ is the d-dimensional training sample set. For a given nonlinear mapping Φ, the original data is projected into empirical feature space $\mathbb{F}$:

$$\Phi : \mathbf{x}_i \in \mathbb{R}^d \longrightarrow \Phi(\mathbf{x}) \in \mathbb{F}. \tag{14}$$

Using Mercer's theorem [19], the inner product of two transformed vectors $\Phi(\mathbf{x})$ in the nonlinear space can be expressed as

$$k(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle, \tag{15}$$

where the operator $\langle\cdot\rangle$ means inner product. Such kernel function is usually called Mercer kernel, and some commonly used Mercer kernels are shown as follows [20]:

$$\begin{gathered} \text{linear kernel: } k(\mathbf{x}, \mathbf{y}) = \mathbf{x}^{\mathrm{T}}\mathbf{y}, \\ \text{Gaussian kernel: } k(\mathbf{x}, \mathbf{y}) = \exp\left(\frac{-\|\mathbf{x}-\mathbf{y}\|^2}{r^2}\right), \\ \text{polynomial kernel: } k(\mathbf{x}, \mathbf{y}) = (\mathbf{x} \cdot \mathbf{y} + d)^l. \end{gathered} \tag{16}$$

Among them, Gaussian kernel is one of the most widespread kernels. However, Gaussian kernel can only reflect the local nonlinear feature of the data, while the linear kernel and polynomial kernel are overall kernel functions. It has been shown that the kernel-based feature extraction algorithm is appropriate for solving the nonlinear problems in face recognition. Nevertheless, the disadvantage of single kernel-based algorithms is lack of the generalization representation capability for multidimensional and multiclass data. Recent applications have indicated that MKL could provide us a more flexible framework to fuse information from different data source and enhance the performance of classifiers [21–26].

In the MKL framework, given M basic kernel functions $\{k_m\}_{m=1}^M$, the multiple kernel function can be generally represented as [27]

$$\begin{gathered} K_{\text{multiple}}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{m=1}^{M} \theta_m k_m(\mathbf{x}_i, \mathbf{x}_j), \\ \text{s.t. } \theta_m \ge 0, \ \sum_{m=1}^{M} \theta_m = 1, \end{gathered} \tag{17}$$

where the weighted coefficient θ_m is commonly obtained by solving the optimal object function of the kernel subspace learning algorithm. It is noticed that optimizing the coefficient θ_m is a critical problem for improving the performance of MKL.

3. The Proposed Multiple Data-Dependent Kernel (MDK)

As mentioned above, given a training data $\mathbf{S}$, the elements of MDK can be formulated as

$$K_{\text{MDDK}}(\mathbf{x}, \mathbf{x}') = q(\mathbf{x}) q(\mathbf{x}') \sum_{i=1}^{I} \gamma_i k_i'(\mathbf{x}, \mathbf{x}'), \tag{18}$$

where $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^d$, $k_i'(\mathbf{x}, \mathbf{x}')$ is the ith basic kernel chosen from the commonly used ones such as Gaussian kernel or polynomial kernel, I is the number of candidate basic kernel, γ_i is the weight for the ith basic kernel, and $q(\cdot)$ is the factor function called data-dependent kernel (DK) which takes the form of

$$q(\mathbf{x}) = \alpha_0 + \sum_{i=1}^{m} \alpha_i k_{\text{Gaussian}}(\mathbf{x}, \mathbf{x}_{\text{ec}}), \tag{19}$$

where α_i, $i = 0, 1, \ldots, m$ is the combination coefficient. The set $\{\mathbf{x}_{\text{ec}} \in \mathbb{R}^d\}$, called the "empirical cores," are chosen from the training data. It is notable that MDK also satisfies the Mercer condition, since $K_{\text{MDDK}}(\mathbf{x}, \mathbf{x}')$ is equal to multiply DK and basic kernels together, which is the linear combination of kernels.

As mentioned above, we can see that the main problems in MDK are to choose the optimal weight γ_i for basic kernels $k_i'(\mathbf{x}, \mathbf{x}')$ and the coefficients α_i of data-dependent kernel $q(\mathbf{x})$. In this paper, we adopt the iterative method based on the maximum margin criterion (MMC) and Fisher scalar to optimize weight γ_i and coefficient α_i, respectively. The schematic of MDK is shown in Figure 1.

3.1. Weight Optimization for Multiple Kernels. To gain good performance of MDKFDA for face recognition, learning proper weights of candidate base kernels is illustrated in this

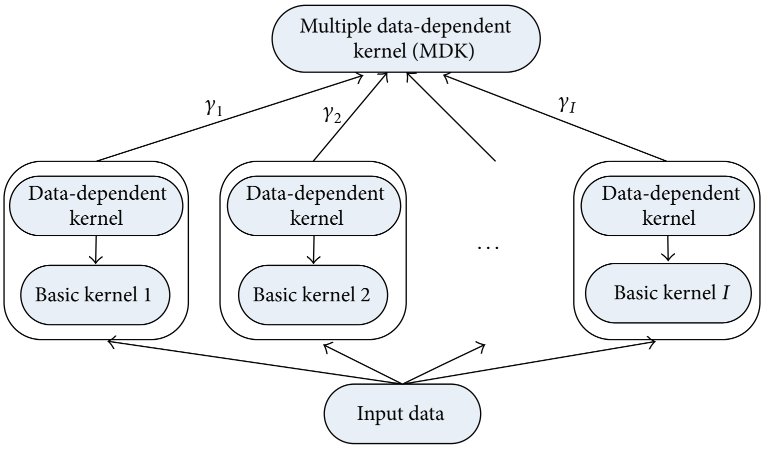


Figure 1: Schematic of the proposed MDK.

section. In KFDA, we measure the class separability in kernel feature space and the kernel Fisher criterion can be expressed as

$$\mathbf{J}^{\Phi}(\mathbf{V}) = \frac{\operatorname{tr}\left(\mathbf{V}^{\mathrm{T}}\mathbf{S}_b^{\Phi}\mathbf{V}\right)}{\operatorname{tr}\left(\mathbf{V}^{\mathrm{T}}\mathbf{S}_w^{\Phi}\mathbf{V}\right)}. \tag{20}$$

In this section, the diagonalization strategy [28] is adopted to find the optimal $\mathbf{V}_{\mathrm{opt}}$, based on which, the maximum margin criterion (MMC) [29] is employed as the objective function to optimize weight γ_i, $i = 1, 2, \ldots, I$. Consider

$$\mathbf{L}(\gamma) = \operatorname{tr}\left({\mathbf{V}_{\mathrm{opt}}}^{\mathrm{T}}\mathbf{S}_b^{\Phi}\mathbf{V}_{\mathrm{opt}}\right) - \operatorname{tr}\left({\mathbf{V}_{\mathrm{opt}}}^{\mathrm{T}}\mathbf{S}_w^{\Phi}\mathbf{V}_{\mathrm{opt}}\right). \tag{21}$$

To maximize $\mathbf{L}(\gamma)$, we introduce a Lagrangian to solve the optimization problem as follows:

$$\mathbb{L}(\gamma, \alpha) = \mathbf{F}(\gamma) + \alpha\left(\sum_{i=1}^{I}\gamma_i - 1\right). \tag{22}$$

A series of partial derivatives can be achieved through differentiating $\mathbb{L}(\gamma, \alpha)$ with respect to $\gamma_1, \gamma_2, \ldots, \gamma_I$ and α. By setting these derivatives to zero, we can use Newton's iteration method to solve these equations, and the optimized weights for multiple kernels are achieved as follows:

$$\tilde{\gamma} = (\gamma_1, \gamma_2, \ldots, \gamma_I, \alpha)^{\mathrm{T}}. \tag{23}$$

3.2. Coefficients Optimization for Data-Dependent Kernel. Since the optimized weights for multiple kernels have been achieved, investigating proper coefficients of DK is described in this section. In empirical feature space, let $\mathbf{J} = \operatorname{tr}(\mathbf{S}_b^{\Phi})/\operatorname{tr}(\mathbf{S}_w^{\Phi})$ denote the Fisher scalar, and $\mathbf{S}_b^{\Phi}$ and $\mathbf{S}_w^{\Phi}$ have been defined in (9) and (10), respectively. Given training dataset $\mathbf{S}$, $\mathbf{K} = [k(\mathbf{x}_i, \mathbf{x}_j)]_{N\times N}$ is the kernel matrix for all samples, whose element can be described as $k_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, $(i, j = 1, 2, \ldots, N)$, and $\mathbf{K}_{pq}$ $(p, q = 1, 2, \ldots, C)$ is the $n_p \times n_q$ submatrices of the $\mathbf{K}$. Hence, $\mathbf{K}$ can be written as

$$\mathbf{K} = \begin{pmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} & \cdots & \mathbf{K}_{1C} \\ \mathbf{K}_{21} & \mathbf{K}_{22} & \cdots & \mathbf{K}_{2C} \\ \vdots & \vdots & \cdots & \vdots \\ \mathbf{K}_{C1} & \mathbf{K}_{C2} & \cdots & \mathbf{K}_{CC} \end{pmatrix}. \tag{24}$$

Consequentially, the between-class scatter matrix $\mathbf{B}$ and within-class scatter matrix $\mathbf{W}$ can be expressed as follows:

$$\begin{aligned} \mathbf{B} &= \begin{bmatrix} \frac{1}{l_1}\mathbf{K}_{11} & 0 & \cdots & 0 \\ 0 & \frac{1}{l_2}\mathbf{K}_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{l_C}\mathbf{K}_{CC} \end{bmatrix} \\ &\quad - \begin{bmatrix} \frac{1}{N}\mathbf{K}_{11} & \frac{1}{N}\mathbf{K}_{12} & \cdots & \frac{1}{N}\mathbf{K}_{1c} \\ \frac{1}{N}\mathbf{K}_{21} & \frac{1}{N}\mathbf{K}_{22} & \cdots & \frac{1}{N}\mathbf{K}_{2C} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{1}{N}\mathbf{K}_{C1} & \frac{1}{N}\mathbf{K}_{C2} & \cdots & \frac{1}{N}\mathbf{K}_{CC} \end{bmatrix}, \\ \mathbf{W} &= \begin{bmatrix} k_{11} & 0 & \cdots & 0 \\ 0 & k_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & k_{NN} \end{bmatrix} \\ &\quad - \begin{bmatrix} \frac{1}{l_1}\mathbf{K}_{11} & 0 & \cdots & 0 \\ 0 & \frac{1}{l_2}\mathbf{K}_{22} & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \frac{1}{l_C}\mathbf{K}_{CC} \end{bmatrix}. \end{aligned} \tag{25}$$

For the multiple kernel $K_{\text{multiple}}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{m=1}^{M} \theta_m k_m(\mathbf{x}_i, \mathbf{x}_j)$, $\mathbf{B}$ and $\mathbf{W}$ can be replaced by the $\mathbf{B}_0$ and the $\mathbf{W}_0$, respectively, and their corresponding kernel matrix is translated into $\mathbf{K} = \mathbf{Q}K_{\text{multiple}}\mathbf{Q}$, in which $\mathbf{Q} = \operatorname{diag}(q(\mathbf{x}_1), q(\mathbf{x}_2), \ldots, q(\mathbf{x}_N))$ and $q(\mathbf{x})$ have been defined in (19), and then we have

$$\mathbf{q} = \begin{bmatrix} 1 & k_1(\mathbf{x}_1, \mathbf{a}_1) & \cdots & k_1(\mathbf{x}_1, \mathbf{a}_m) \\ 1 & k_1(\mathbf{x}_2, \mathbf{a}_1) & \cdots & k_1(\mathbf{x}_2, \mathbf{a}_m) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & k_1(\mathbf{x}_N, \mathbf{a}_1) & \cdots & k_1(\mathbf{x}_N, \mathbf{a}_m) \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_m \end{bmatrix} \triangleq \mathbf{K}_1 \boldsymbol{\alpha}. \quad (26)$$

Theorem 1. *Let $\mathbf{1}_k$ be the k-dimensional vector with unity elements; the Fisher scalar $\mathbf{J}$ is equivalent to*

$$\mathbf{J} = \frac{\mathbf{1}_N^{\mathrm{T}} \mathbf{B} \mathbf{1}_N}{\mathbf{1}_N^{\mathrm{T}} \mathbf{W} \mathbf{1}_N} = \frac{\mathbf{q}^{\mathrm{T}} \mathbf{B}_0 \mathbf{q}}{\mathbf{q}^{\mathrm{T}} \mathbf{W}_0 \mathbf{q}}. \quad (27)$$

Proof. As shown in Section 2.1, the dimension of empirical feature space is set as p $(p < N)$. Then, $\mathbf{H}$ and $\mathbf{H}_i$, $i = 1, \ldots, C$, are, respectively, defined as the $N \times p$ and $l_i \times p$, $i = 1, \ldots, C$ matrices whose rows are the vectors $\{\mathbf{h}_j^{\mathrm{T}}\}$, $j = 1, 2, \ldots, N$ and $\{\mathbf{h}_j^{\mathrm{T}}\}$, $j = 1, 2, \ldots, l_i$. The $\mathbf{m}_0^{\Phi}$ and $\mathbf{m}_i^{\Phi}$ in Section 2.1 can be expressed as follows:

$$\begin{aligned} \mathbf{m}_0^{\Phi} &= \frac{1}{N} \sum_{j=1}^{N} \mathbf{h}_j = \frac{1}{N} \mathbf{H}^{\mathrm{T}} \mathbf{I}_N, \\ \mathbf{m}_i^{\Phi} &= \frac{1}{l_i} \sum_{j=l_{i-1}+1}^{l_1+\cdots+l_i} \mathbf{h}_j = \frac{1}{l_i} \mathbf{H}_i^{\mathrm{T}} \mathbf{I}_{l_i}, \quad i = 1, \ldots, C. \end{aligned} \quad (28)$$

Moreover, since the empirical feature space maintains the dot product, (24) is equivalent to

$$\mathbf{K} = \mathbf{H}\mathbf{H}^{\mathrm{T}} = \begin{bmatrix} \mathbf{H}_1 & \mathbf{H}_2 & \cdots & \mathbf{H}_C \end{bmatrix} \begin{bmatrix} \mathbf{H}_1^{\mathrm{T}} \\ \mathbf{H}_2^{\mathrm{T}} \\ \vdots \\ \mathbf{H}_C^{\mathrm{T}} \end{bmatrix}, \quad (29)$$

where $\mathbf{K}_{ij} = \mathbf{H}_i \mathbf{H}_j^{\mathrm{T}}$, $i, j = 1, 2, \ldots, C$. As such, $\operatorname{tr}(\mathbf{S}_b^{\Phi})$ and $\operatorname{tr}(\mathbf{S}_w^{\Phi})$ can be described as

$$\begin{aligned}
\operatorname{tr}\left(\mathbf{S}_b^{\Phi}\right) &= \frac{1}{N} \sum_{i=1}^{C} l_i \left(\mathbf{m}_i^{\Phi} - \mathbf{m}_0^{\Phi}\right)^{\mathrm{T}} \left(\mathbf{m}_i^{\Phi} - \mathbf{m}_0^{\Phi}\right) \\
&= \frac{1}{N} \sum_{i=1}^{C} l_i \mathbf{m}_i^{\Phi\mathrm{T}} \mathbf{m}_i^{\Phi} - \mathbf{m}_0^{\Phi\mathrm{T}} \mathbf{m}_0^{\Phi} \\
&= \frac{1}{N} \sum_{i=1}^{C} \frac{1}{l_i} \mathbf{I}_{l_i}^{\mathrm{T}} \mathbf{H}_i \mathbf{H}_i^{\mathrm{T}} \mathbf{I}_{l_i} - \frac{1}{N^2} \mathbf{I}_N^{\mathrm{T}} \mathbf{H}\mathbf{H}^{\mathrm{T}} \mathbf{I}_N \\
&= \frac{1}{N} \sum_{i=1}^{C} \frac{1}{l_i} \mathbf{I}_{l_i}^{\mathrm{T}} \mathbf{K}_{ii} \mathbf{I}_{l_i} - \frac{1}{N^2} \mathbf{I}_N^{\mathrm{T}} \mathbf{K} \mathbf{I}_N \\
&= \frac{1}{N} \begin{bmatrix} \mathbf{I}_{l_1}^{\mathrm{T}} & \mathbf{I}_{l_2}^{\mathrm{T}} & \cdots & \mathbf{I}_{l_c}^{\mathrm{T}} \end{bmatrix} \\
&\quad \times \begin{bmatrix} \frac{1}{l_1}\mathbf{K}_{11} & 0 & \cdots & 0 \\ 0 & \frac{1}{l_2}\mathbf{K}_{22} & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \frac{1}{l_c}\mathbf{K}_{cc} \end{bmatrix} \begin{bmatrix} \mathbf{I}_{l_1}^{\mathrm{T}} \\ \mathbf{I}_{l_2}^{\mathrm{T}} \\ \vdots \\ \mathbf{I}_{l_c}^{\mathrm{T}} \end{bmatrix} \\
&\quad - \frac{1}{N} \mathbf{I}_N^{\mathrm{T}} \begin{bmatrix} \frac{1}{N}\mathbf{K}_{11} & \frac{1}{N}\mathbf{K}_{12} & \cdots & \frac{1}{N}\mathbf{K}_{1c} \\ \frac{1}{N}\mathbf{K}_{21} & \frac{1}{N}\mathbf{K}_{22} & \cdots & \frac{1}{N}\mathbf{K}_{2c} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{1}{N}\mathbf{K}_{c1} & \frac{1}{N}\mathbf{K}_{c2} & \cdots & \frac{1}{N}\mathbf{K}_{cc} \end{bmatrix} \\
&= \frac{1}{N} \mathbf{I}_N^{\mathrm{T}} \mathbf{B} \mathbf{I}_N, \\
\operatorname{tr}\left(\mathbf{S}_w^{\Phi}\right) &= \frac{1}{N} \sum_{i=1}^{C} \sum_{j=1}^{l_i} \left(\Phi\left(\mathbf{x}_j^i\right) - \mathbf{m}_i^{\Phi}\right)^{\mathrm{T}} \left(\Phi\left(\mathbf{x}_j^i\right) - \mathbf{m}_i^{\Phi}\right) \\
&= \frac{1}{N} \sum_{i=1}^{C} \left(\sum_{j=1}^{l_i} \left(\Phi\left(\mathbf{x}_j^i\right)^{\mathrm{T}} \Phi\left(\mathbf{x}_j^i\right) \right) - l_i \left(\mathbf{m}_i^{\Phi}\right)^{\mathrm{T}} \mathbf{m}_i^{\Phi} \right) \\
&= \frac{1}{N} \left(\sum_{j=1}^{N} \Phi\left(\mathbf{x}_j\right)^{\mathrm{T}} \Phi\left(\mathbf{x}_j\right) - \sum_{i=1}^{C} \frac{1}{l_i} \mathbf{I}_{l_i}^{\mathrm{T}} \mathbf{H}_i \mathbf{H}_i^{\mathrm{T}} \mathbf{I}_{l_i} \right) \\
&= \frac{1}{N} \left(\sum_{j=1}^{N} k_{jj} - \begin{bmatrix} \mathbf{I}_{l_1}^{\mathrm{T}} & \mathbf{I}_{l_2}^{\mathrm{T}} & \cdots & \mathbf{I}_{l_c}^{\mathrm{T}} \end{bmatrix} \right. \\
&\quad \left. \times \begin{bmatrix} \frac{1}{l_1}\mathbf{K}_{11} & 0 & \cdots & 0 \\ 0 & \frac{1}{l_2}\mathbf{K}_{22} & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \frac{1}{l_c}\mathbf{K}_{cc} \end{bmatrix} \begin{bmatrix} \mathbf{I}_{l_1}^{\mathrm{T}} \\ \mathbf{I}_{l_2}^{\mathrm{T}} \\ \vdots \\ \mathbf{I}_{l_c}^{\mathrm{T}} \end{bmatrix} \right) \\
&= \frac{1}{N} \mathbf{I}_N^{\mathrm{T}} \mathbf{W} \mathbf{I}_N.
\end{aligned} \quad (30)$$

From the above description, it is remarkable that $\mathbf{B} = \mathbf{Q}\mathbf{B}_0\mathbf{Q}$ and $\mathbf{W} = \mathbf{Q}\mathbf{W}_0\mathbf{Q}$, where $\mathbf{Q}\mathbf{I}_N = \mathbf{q}$. Finally, the relationship is proved:

$$\mathbf{J} = \frac{\operatorname{tr}\left(\mathbf{S}_b^{\Phi}\right)}{\operatorname{tr}\left(\mathbf{S}_w^{\Phi}\right)} = \frac{\mathbf{1}_N^{\mathrm{T}}\mathbf{B}\mathbf{1}_N}{\mathbf{1}_N^{\mathrm{T}}\mathbf{W}\mathbf{1}_N} = \frac{\mathbf{q}^{\mathrm{T}}\mathbf{B}_0\mathbf{q}}{\mathbf{q}^{\mathrm{T}}\mathbf{W}_0\mathbf{q}}. \qquad (31)$$

In order to obtain the optimal coefficients, $\mathbf{J}$ should be maximized. Since $\mathbf{q} = \mathbf{K}_1\boldsymbol{\alpha}$, $\mathbf{J}$ can be reformulated as

$$\mathbf{J}(\boldsymbol{\alpha}) = \frac{\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{K}_1^{\mathrm{T}}\mathbf{B}_0\mathbf{K}_1\boldsymbol{\alpha}}{\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{K}_1^{\mathrm{T}}\mathbf{W}_0\mathbf{K}_1\boldsymbol{\alpha}}, \qquad (32)$$

where $\mathbf{M}_0 = \mathbf{K}_1^{\mathrm{T}}\mathbf{B}_0\mathbf{K}_1$ and $\mathbf{N}_0 = \mathbf{K}_1^{\mathrm{T}}\mathbf{W}_0\mathbf{K}_1$. To maximize $\mathbf{J}(\boldsymbol{\alpha})$, the standard gradient approach is adopted. Given that

$$\begin{aligned}\mathbf{J}_1(\boldsymbol{\alpha}) &= \boldsymbol{\alpha}^{\mathrm{T}}\mathbf{K}_1^{\mathrm{T}}\mathbf{B}_0\mathbf{K}_1\boldsymbol{\alpha},\\ \mathbf{J}_2(\boldsymbol{\alpha}) &= \boldsymbol{\alpha}^{\mathrm{T}}\mathbf{K}_1^{\mathrm{T}}\mathbf{W}_0\mathbf{K}_1\boldsymbol{\alpha},\end{aligned} \qquad (33)$$

respectively, the partial differential of $\mathbf{J}_1(\boldsymbol{\alpha})$ and $\mathbf{J}_2(\boldsymbol{\alpha})$ with respect to $\boldsymbol{\alpha}$ are

$$\begin{aligned}\frac{\partial\mathbf{J}_1(\boldsymbol{\alpha})}{\partial\boldsymbol{\alpha}} &= 2\mathbf{K}_1^{\mathrm{T}}\mathbf{B}_0\mathbf{K}_1\boldsymbol{\alpha},\\ \frac{\partial\mathbf{J}_2(\boldsymbol{\alpha})}{\partial\boldsymbol{\alpha}} &= 2\mathbf{K}_1^{\mathrm{T}}\mathbf{W}_0\mathbf{K}_1\boldsymbol{\alpha},\end{aligned} \qquad (34)$$

and the partial differential of $\mathbf{J}(\boldsymbol{\alpha})$ is

$$\frac{\partial\mathbf{J}(\boldsymbol{\alpha})}{\partial\boldsymbol{\alpha}} = \frac{2}{\mathbf{J}_2^2}\left(\mathbf{J}_2\mathbf{K}_1^{\mathrm{T}}\mathbf{B}_0\mathbf{K}_1 - \mathbf{J}_1\mathbf{K}_1^{\mathrm{T}}\mathbf{W}_0\mathbf{K}_1\right)\boldsymbol{\alpha}. \qquad (35)$$

Let $\partial\mathbf{J}(\boldsymbol{\alpha})/\partial\boldsymbol{\alpha} = 0$; it is obtained that

$$\begin{aligned}\mathbf{J}_1\mathbf{K}_1^{\mathrm{T}}\mathbf{W}_0\mathbf{K}_1\boldsymbol{\alpha} &= \mathbf{J}_2\mathbf{K}_1^{\mathrm{T}}\mathbf{B}_0\mathbf{K}_1\boldsymbol{\alpha},\\ \mathbf{J}\boldsymbol{\alpha} &= \left(\mathbf{K}_1^{\mathrm{T}}\mathbf{W}_0\mathbf{K}_1\right)^{-1}\left(\mathbf{K}_1^{\mathrm{T}}\mathbf{B}_0\mathbf{K}_1\right)\boldsymbol{\alpha}, \quad \mathbf{K}_1^{\mathrm{T}}\mathbf{W}_0\mathbf{K}_1 \neq 0.\end{aligned} \qquad (36)$$

Hence, $\mathbf{J}$ is equal to the largest eigenvalue of the matrix $(\mathbf{K}_1^{\mathrm{T}}\mathbf{W}_0\mathbf{K}_1)^{-1}(\mathbf{K}_1^{\mathrm{T}}\mathbf{B}_0\mathbf{K}_1)$, and the optimal $\boldsymbol{\alpha}$ is the eigenvector corresponding to the largest eigenvalue; thus, an iteration algorithm is employed to calculate the optimal $\boldsymbol{\alpha}$:

$$\boldsymbol{\alpha}^{(m+1)} = \boldsymbol{\alpha}^{(m)} + \varepsilon\left(\frac{1}{\mathbf{J}_2}\mathbf{K}_1^{\mathrm{T}}\mathbf{B}_0\mathbf{K}_1 - \frac{\mathbf{J}}{\mathbf{J}_2}\mathbf{K}_1^{\mathrm{T}}\mathbf{W}_0\mathbf{K}_1\right)\boldsymbol{\alpha}^{(m)}, \qquad (37)$$

where ε is the learning rate, and it is given by

$$\varepsilon(t) = \varepsilon_0\left(1 - \frac{t}{N}\right), \qquad (38)$$

where ε_0 is the initial learning rate, t, and N denotes the current iteration number and prespecified iteration number, respectively. In summary, the optimal coefficients can be obtained by choosing ε_0 and N properly. □

Figure 2: Sample images of one person in the ORL database.

Figure 3: Samples images of one person in the YALE database.

3.3. Complete MDKFDA Algorithm. In summary of the discussion so far, the steps of complete MDKFDA algorithm are described as follows.

Step 1 (construct the MDK). Gaussian kernel is adopted to construct the data-dependent kernel (DK), while the linear kernel, Gaussian kernel, and polynomial kernel are employed as the base kernels of multiple kernel function.

Step 2 (optimize the weights). The maximum margin criterion (MMC) is employed as the objective function to optimize weights for multiple kernels, and the optimized coefficients for DK are achieved by virtue of the Fisher scalar.

Step 3 (transform the data). The MDK is used to transform the input space $\mathbb{R}^d$ into feature space $\mathbb{R}^p$, by which the original input data $\{\mathbf{x}_i \mid \mathbf{x}_i \in \mathbb{R}^d\}_{i=1}^N$ is converted into feature data $\{\mathbf{y}_i \mid \mathbf{y}_i \in \mathbb{R}^p\}_{i=1}^T$. The transformation is $\mathbf{P} = (\mathbf{w}_1, \ldots, \mathbf{w}_t)$.

Step 4 (extract the Fisher discriminant vectors). $\mathbf{S}_b^{\Phi}$, $\mathbf{S}_w^{\Phi}$ in $\mathbb{R}^p$ are calculated to get the Fisher criterion $\mathbf{J}^{\Phi}(\mathbf{V})$. By maximizing $\mathbf{J}^{\Phi}(\mathbf{V})$, the Fisher optimal discriminant vectors are achieved, and the Fisher discriminant transformation is $\mathbf{G} = (\mathbf{V}_1, \ldots, \mathbf{V}_d)$.

Step 5 (obtain the MDKFDA feature vector). Based on the first four steps, the expression of the MDKFDA feature vector $\mathbf{z} = \mathbf{G}^{\mathrm{T}}\mathbf{P}^{\mathrm{T}}\Phi(\mathbf{x})$ is obtained.

4. Experimental Results and Discussions

In this section, we conduct several experiments on three face databases to evaluate the performance of the proposed MDKFDA algorithm by comparing it with several widespread algorithms in face recognition, including PCA, LPP, FDA, KPCA, KFDA, and DKFDA. The ORL face database [30], YALE face database [31], and PIE face database [32] are adopted in the experiments, and partial sample images of one person from different databases are shown in Figures 2, 3, and 4. In the following experiments, we select randomly 5 images per individual as training set and the rest 5 for testing. To make the experiments more reasonable, we repeated the trails 10 times to achieve an average performance.

Table 1: Comparison of recognition ratio between linear algorithms and MDKFDA.

Algorithm	PCA	LPP	LDA	MDKFDA
Recognition ratio (%)	82.0	75.5	74.3	95.8

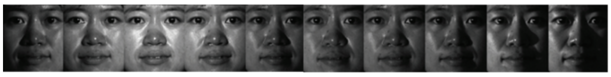


Figure 4: Sample images of one person in the PIE database.

4.1. Face Recognition Using ORL Database. The ORL face database contains 400 face images of 40 individuals, and variations in these 400 face images include angle, lighting, expression, and face details. As shown in Figure 2, the size of all the original images were shaped into 48 ∗ 48 pixels, and the primary part of the original image was reserved. Three kernels are employed as the base kernels of multiple kernel function in MDK, including linear kernel, Gaussian kernel, and polynomial kernel. Moreover, Gaussian kernel is adopted to construct DK. Table 1 shows the comparison of maximal average recognition ratio between several nonkernel algorithms and the proposed MDKFDA algorithm. Table 2 reports the comparison of maximal average recognition ratio between several single kernel-based algorithms and the proposed MDKFDA. According to the experiment results, the MDKFDA algorithm outperforms the other algorithms, which implies that the MDKFDA can integrate multiple base kernels with data-dependent kernel (DK) effectively to improve the recognition ratio. Besides, it can be seen that all the single kernel-based feature extraction algorithms outperform their corresponding linear versions, which indicates that the kernel-based algorithm is advantageous for face recognition.

4.2. Face Recognition Using YALE Database. The YALE database contains 165 front view face images from 15 individuals. Each individual has eleven images that vary with expression and configurations. Three kernels are employed as the base kernels of multiple kernel function in MDK, including linear kernel, Gaussian kernel, and polynomial kernel. Moreover, Gaussian kernel is adopted to construct DK. Table 3 displays the average recognition rates of PCA, LDA, LPP, and MDKFDA. Table 4 reports the performance comparison of different kernel-based algorithms, including KPCA, KFDA, DKFDA, and MDKFDA. The results indicate that the MDKFDA algorithm outperforms the other algorithms and all the single kernel-based feature extraction algorithms outperform their corresponding linear versions.

4.3. Face Recognition Using PIE Database. The PIE face database contains over 40,000 face images from 68 people, and the images are captured under 13 different poses, 43 different illumination, and 4 different expressions. In this test, we select 150 face images from 15 individuals. The selection of relevant base kernels for multiple kernel function and DK is the same as that in the former experiments. From Tables 5 and 6, it can be seen that the MDKFDA algorithm outperforms the other algorithms.

Table 2: Comparison of recognition ratio between single kernel-based algorithms and MDKFDA.

Algorithm	Gaussian kernel (%)	Polynomial kernel (%)
KPCA	86.5	84.5
KFDA	88.0	85.0
DKFDA	93.5	92.5
MDKFDA	95.8	

Table 3: Comparison of recognition ratio between linear algorithms and MDKFDA.

Algorithm	PCA	LPP	LDA	MDKFDA
Recognition ratio (%)	78.7	73.3	81.3	96.1

Table 4: Comparison of recognition ratio between single kernel-based algorithms and MDKFDA.

Algorithm	Gaussian kernel (%)	Polynomial kernel (%)
KPCA	80.5	80.0
KFDA	90.7	86.7
DKFDA	93.3	93.3
MDKFDA	96.1	

Table 5: Comparison of recognition ratio between linear algorithms and MDKFDA.

Algorithm	PCA	LPP	LDA	MDKFDA
Recognition ratio (%)	50.67	50.67	65.2	94.9

Table 6: Comparison of recognition ratio between single kernel-based algorithms and MDKFDA.

Algorithm	Gaussian kernel (%)	Polynomial kernel (%)
KPCA	74.7	50.7
KFDA	81.3	78.7
DKFDA	93.5	90.3
MDKFDA	94.9	

4.4. Discussions. Experiments based on the three face databases have been systematically implemented, and the results reveal some interesting findings which are summarized as follows.

(1) The single kernel-based nonlinear feature extraction algorithms such as KPCA, KFDA, and DKFDA perform better than their corresponding linear versions such as PCA and LDA. The main reason is attributed to the fact that, compared to the linear techniques, the features extracted by kernel-based algorithms can better describe the complex and nonlinear variations of face images, that is, illumination, pose, and facial expression. Hence, a better recognition rate can be achieved.

(2) The average recognition rates of PCA, LDA, and LPP on PIE database are significantly less than those on the other two databases. The reason is mainly as follows. Since the images from the PIE database are captured under more complicated conditions, they appear to have more complicated nonlinear characteristics than those from the other databases, which makes them difficult to handle by linear algorithms.

(3) The DKFDA algorithm outperforms the other single kernel-based algorithms, that is, KPCA, KLPP, and KFDA. As shown in Table 6, when KPCA with polynomial kernel is adopted, the recognition ratio of face images is not increased. This is because the characteristics of kernel are ill-suited for some database. However, the DK can solve this problem, and the reason is that the DKFDA algorithm has the adaptability for different databases. The structure of DK can be changed by adjusting the kernel parameter using iterative method, so various input data can be better expressed.

(4) The proposed MDKFDA algorithm consistently performs better than the KPCA, KLPP, KFDA, and DKFDA as well as their corresponding linear version, which indicates that, compared to the single kernel-based algorithms, the MDKFDA algorithm can effectively integrate the multiple base kernels with data-dependent kernel (DK) and gain a good performance on face recognition.

5. Conclusions

In this paper, on the assumption that multiple kernel-based recognition algorithms can depict the complex and heterogeneous face image dataset by the utilization of multiple descriptors, a novel kernel-based approach for face recognition, called multiple data-dependent kernel Fisher discriminant analysis (MDKFDA), is proposed in this paper. Focusing on the construction of MDK, two main issues have been considered. The first issue concerns optimizing the weights of multiple base kernels. For this purpose, by maximizing the margin maximization criterion (MMC), an iterative method based on Lagrange multipliers is adopted to yield the optimized weights. The second issue aims at optimizing the coefficients of data-dependent kernel. To this end, by solving the optimization equation based on Fisher scalar, a gradient-based learning algorithm is employed to yield the optimized coefficients. Finally, the resulting multiple kernel functions and data-dependent kernel are integrated together as a new kernel, which is incorporated into the KFDA to construct the MDKFDA. Experiments on three face databases prove the effectiveness of the MDKFDA, and this algorithm is ready to be applied to other classification applications in the future.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (no. 51374099), Heilongjiang Province Natural Science Foundation for the Youth (no. QC2012C070), Heilongjiang Province Natural Science Foundation (F201345), and the Fundamental Research Funds for the Central Universities of China (no. HEUCF140807).

References

[1] J. Yang, G. Cheng, and M. Li, “Extraction of affine invariant features using fractal,” *Advances in Mathematical Physics*, vol. 2013, Article ID 950289, 8 pages, 2013.

[2] Z. Li, J. Yang, M. Li, and R. Lan, “Estimation of large scalings in images based on multilayer pseudopolar fractional fourier transform,” *Mathematical Problems in Engineering*, vol. 2013, Article ID 179489, 9 pages, 2013.

[3] D. L. Swets, “Using discriminant eigenfeatures for image retrieval,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 831–836, 1996.

[4] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, “Eigenfaces vs. fisherfaces: recognition using class specific linear projection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.

[5] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, Wiley-Interscience, Hoboken, NJ, USA, 2nd edition, 2000.

[6] X. He and P. Niyogi, “Locality preserving projections,” *Neural Information Processing Systems*, vol. 16, pp. 435–445, 2003.

[7] B. Scholkopf, A. Smola, and K.-R. Muller, “Kernel principal component analysis,” in *Advance in Kernel Methods, Support Vector Learning*, B. Scholkopf, C. J. C. Burges, and A. J. Smola, Eds., MIT Press, Cambridge, Mass, USA, 1999.

[8] S. Mika, G. Rätsch, J. Weston, B. Schölkopf, and K.-R. Müller, “Fisher discriminant analysis with kernels,” in *Proceedings of the IEEE Signal Processing Society Workshop on Neural Networks for Signal Processing*, pp. 41–48, August 1999.

[9] S. Zafeiriou, G. Tzimiropoulos, M. Petrou, and T. Stathaki, “Regularized kernel discriminant analysis with a robust kernel for face recognition and verification,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 3, pp. 526–534, 2012.

[10] Z. Wang and X. Sun, “Manifold adaptive kernel local fisher discriminant analysis for face recognition,” *Journal of Multimedia*, vol. 7, no. 6, pp. 387–393, 2012.

[11] A. Rakotomamonjy, F. R. Bach, S. Canu, and Y. Grandvalet, “Simple MKL,” *Journal of Machine Learning Research*, vol. 9, pp. 2491–2521, 2008.

[12] D. P. Lewis, T. Jebara, and W. S. Noble, “Nonstationary kernel combination,” in *Proceedings of the 23rd International Conference on Machine Learning (ICML '06)*, pp. 553–560, Pittsburgh, Pa, USA, June 2006.

[13] M. Gönen and E. Alpaydin, “Localized multiple kernel learning,” in *Proceedings of the 25th International Conference on Machine Learning*, pp. 352–359, ACM, Helsinki, Finland, July 2008.

[14] S. C. H. Hoi, R. Jin, P. Zhao, and T. Yang, “Online multiple kernel classification,” *Machine Learning*, vol. 90, no. 2, pp. 289–316, 2013.

[15] S. Amari and S. Wu, “Improving support vector machine classifiers by modifying kernel functions,” *Neural Networks*, vol. 12, no. 6, pp. 783–789, 1999.

[16] H. Xiong, M. N. S. Swamy, and M. O. Ahmad, "Optimizing the kernel in the empirical feature space," *IEEE Transactions on Neural Networks*, vol. 16, no. 2, pp. 460–474, 2005.

[17] I.-L. Chen, C.-H. Li, B.-C. Kuo, and H.-Y. Huang, "Applying optimal algorithm to data-dependent kernel for hyperspectral image classification," in *Proceedings of the 30th IEEE International Geoscience and Remote Sensing Symposium (IGARSS '10)*, pp. 2808–2811, Taichung, Taiwan, July 2010.

[18] J. Yang, A. F. Frangi, D. Zhang, and Z. Jin, "KPCA plus LDA: a complete kernel fisher discriminant framework for feature extraction and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 2, pp. 230–244, 2005.

[19] J. Mercer, "Functions of positive and negative type and their connection with the theory of integral equations," *Philosophical Transactions of the Royal Society A*, vol. 209, no. 441–458, pp. 415–446, 1909.

[20] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, Cambridge, Mass, USA, 2002.

[21] F. R. Bach, G. R. G. Lanckriet, and M. I. Jordan, "Multiple kernel learning, conic duality, and the SMO algorithm," in *Proceedings of the 21st International Conference on Machine Learning (ICML '04)*, pp. 41–48, July 2004.

[22] T. Sun, L. Jiao, F. Liu, S. Wang, and J. Feng, "Selective multiple kernel learning for classification with ensemble strategy," *Pattern Recognition*, vol. 46, no. 11, pp. 3081–3090, 2013.

[23] C. Yeh, C. Huang, and S. Lee, "Multi-kernel support vector clustering for multi-class classification," *International Journal of Innovative Computing, Information and Control*, vol. 6, no. 5, pp. 2245–2262, 2010.

[24] X. Liu, L. Wang, J. Yin, E. Zhu, and J. Zhang, "An efficient approach to integrating radius information into multiple kernel learning," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 43, no. 2, pp. 557–569, 2013.

[25] H. Huang, Y. Chuang, and C. Chen, "Multiple kernel fuzzy clustering," *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 1, pp. 120–134, 2012.

[26] Y. Gu, C. Wang, D. You, Y. Zhang, S. Wang, and Y. Zhang, "Representative multiple kernel learning for classification in hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 7, pp. 2852–2865, 2012.

[27] H. Q. Wang, F. C. Sun, Y. N. Cai, N. Chen, and L. G. Ding, "On multiple kernel learning methods," *Acta Automatica Sinica*, vol. 36, no. 8, pp. 1037–1050, 2010.

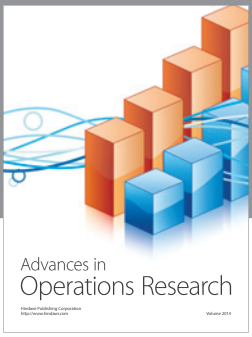
[28] J. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "Face recognition using kernel direct discriminant analysis algorithms," *IEEE Transactions on Neural Networks*, vol. 14, no. 1, pp. 117–126, 2003.

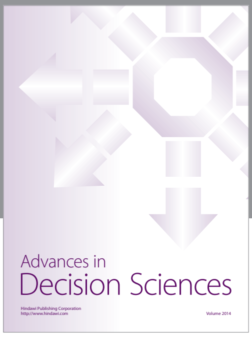
[29] H. Li, T. Jiang, and K. Zhang, "Efficient and robust feature extraction by maximum margin criterion," in *Advances in Neural Information Processing Systems 16*, S. Thrun, L. Saul, and B. Schölkopf, Eds., pp. 157–165, MIT Press, Cambridge, Mass, USA, 2004.

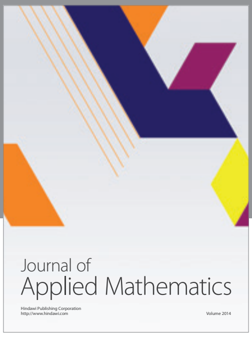
[30] ORL Face Database AT&T Laboratories, Cambridge, UK, http://www.cl.cam.ac.uk/research/dtg/attarchive/facesataglance.html.

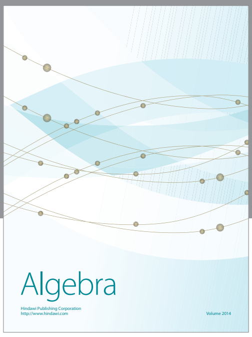
[31] *Yale Face Database: UCSD Computer Vision, America, USA*, http://vision.ucsd.edu/content/yale-face-database.

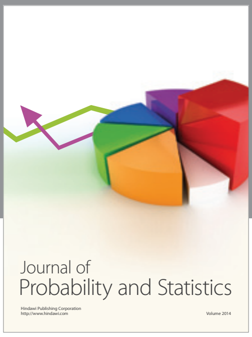
[32] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression database," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1615–1618, 2003.


Advances in
Operations Research

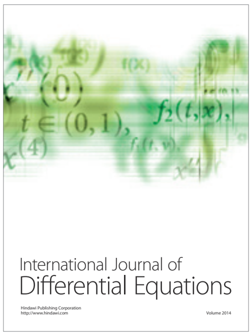

Advances in
Decision Sciences


Journal of
Applied Mathematics

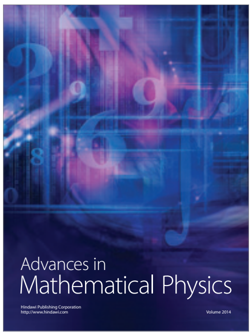

Algebra

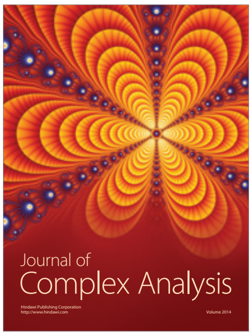

Journal of
Probability and Statistics

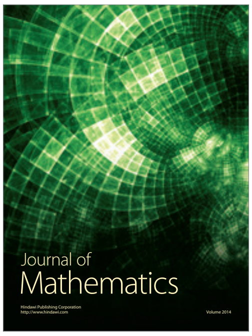
The Scientific
World Journal

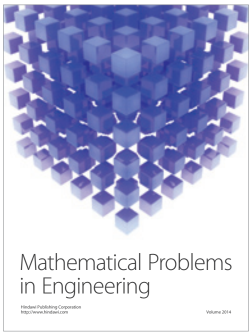

International Journal of
Differential Equations

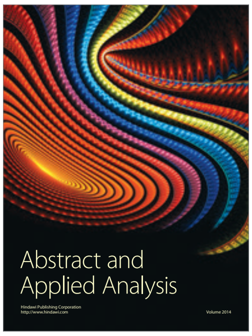
International Journal of
Combinatorics

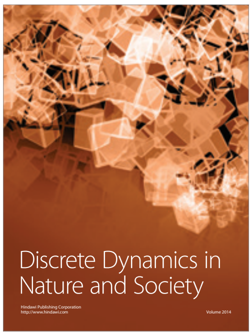

Advances in
Mathematical Physics

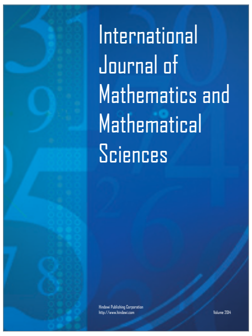

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

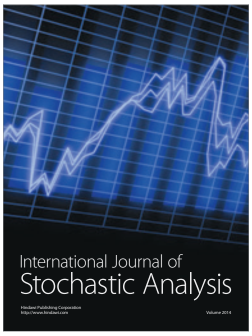

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization