Neural Comput & Applic (2017) 28:3429–3439
DOI 10.1007/s00521-016-2206-y

ORIGINAL ARTICLE

A manifold framework of multiple-kernel learning for hyperspectral image classification

Xiaodan Xie[1] · Bohu Li[1] · Xudong Chai[2]

Received: 4 May 2015 / Accepted: 16 January 2016 / Published online: 9 March 2016

Abstract Manifold learning is a promising intelligent data analysis method, and the manifold learning preserves the local embedding features of the data in manifold mapping space. Manifold learning has its limitations on extracting the nonlinear features of the data in many applications. For example, hyperspectral image classification needs to seek the nonlinear local relationships between spectral curves. For that, researchers applied the kernel trick to manifold learning in the previous works. The kernel-based manifold learning was developed, but still endures the problem that the inappropriate kernel model reduces the system performance. In order to solve the problem of kernel model selection, we propose a manifold framework of multiple-kernel learning for the application of hyperspectral image classification. In this framework, the quasiconformal mapping-based multiple-kernel model is optimized based on the optimization objective equation, which maximizes the class discriminant ability of data. Accordingly, the discriminative structure of data distribution is achieved for classification with the quasiconformal mapping-based multiple-kernel model.

Keywords Manifold learning · Multiple-kernel learning · Hyperspectral image classification

✉ Xiaodan Xie
xiexiaodanbeijing@126.com

Bohu Li
bohulibeijing@126.com

Xudong Chai
xudongchaibeijing@126.com

[1] School of Automation Science and Electrical Engineering, Beihang University, Beijing, China

[2] Beijing Simulation Center, Beijing, China

1 Introduction

Manifold learning promotes dimensionality reduction of intelligent data analysis. Dimensionality reduction is to map a high-dimensional data into a lower-dimensional space with linear transformation matrix, and the data in low-dimensional space are easily analyzed. Manifold learning preserves the nonlinear manifold and constructs a smooth and graded mesh of data. Manifold-based learning seeks the natural geometric dimensionality reduction of data for the excellent classification performance. The most popular methods are principal component analysis (PCA) [1], linear discriminant analysis (LDA) [2], principal curves [3], and principal surfaces [4]. The main manifold learning methods include self-organizing mapping (SOM) [5], visualization-induced SOM (ViSOM) [6], locally linear embedding (LLE) [7], Isomap [8], locality preserving projection (LPP) [9, 10], and class-wise locality preserving projection (CLPP) [11]. These methods have the different criterions of dimensionality reduction as follows. Firstly, SOM learns a nonparametric model with a topological constraint of lines, squares, or hexagonal grids [12, 13]. Secondly, ViSOM constructs a smooth and graded mesh of data as the discrete version of principal curve or surface. Thirdly, LLE preserves the geometrical perspective, and Isomap preserves the geometric relationships and neighborhoods of the data. Finally, LPP locates both training sample and the test data point, and CLPP preserves the local structure of the original data together with the class information.

Kernel CLPP is developed with kernel trick for feature extraction [11]. Researchers present an alternative framework of kernel LPP (KLPP) to develop a framework of KPCA + LPP [5, 8] for image for target recognition, and other improved kernel-based LPP methods were presented in the previous works [6, 13, 14]. As the kernel learning methods, kernel manifold learning still endures the kernel model selection. The geometrical structure of the data distribution is determined by the kernel function. The discriminative ability may be worse under the inappropriate kernel selection [15–17]. Selecting the optimal parameter does not change the geometry structure of data distribution. So, some kernel optimization methods are proposed to improve the performances of kernel learning machines, for example, data-depend kernel [18], kernel-adaptive support vector machine [19–21], sparse multiple-kernel learning [22], large-scale multiple-kernel learning [23], Lp-norm multiple-kernel learning [24].

As the above discussion, the kernel-based manifold learning framework includes two stages: kernel mapping and manifold projection. Multiple-kernel learning methods aim to construct a kernel model with a linear combination of fixed base kernels. Learning the kernel then consists of learning the weighting coefficients for each base kernel. There are two advantages: (1) multiple-kernel learning combines the kernel functions with the different characteristics of the data, so it preserves the nonlinear mapping characteristics of kernel functions; (2) quasiconformal kernel has its ability to change the data structure. So, we propose a manifold framework of multiple-kernel learning for hyperspectral image classification, and the framework is to solve the problems of the determination of the kernel function and its parameters of kernel manifold learning for the practical application system.

2 Proposed scheme

2.1 Motivation

Kernel-based manifold learning is an effective method on the applications of solving the nonlinear problems. As the important indexes of system performances, recognition accuracy or prediction accuracy is largely increased by the nonlinear kernel trick. However, the performance of kernel-based manifold learning system is largely influenced by the function and parameter of kernel. Only optimizing the parameters is not effective, because the data distribution is not changing with the changing of the parameter of kernel function. Researchers have proposed alternative kernel function to solve this problem, for example, multiple kernel and quasiconformal kernel. Firstly, multiple-kernel learning combines the kernel functions with the different characteristics of the data. Accordingly, MKL combines many features and is better to describe the data features than the single feature extraction method. Multiple-kernel learning preserves the nonlinear mapping characteristics of kernel functions. And it shows the possibility of using different kernel functions for kernel-based manifold learning. Secondly, quasiconformal kernel has its ability to change the data structure. It is feasible to improve kernel-based manifold learning through adjusting the quasiconformal.

In this paper, we improve the kernel-based manifold learning through considering enough the advantages of multiple-kernel learning and quasiconformal kernel learning. A manifold framework of multiple-kernel learning is proposed for hyperspectral image classification. Based on the traditional kernel-based manifold learning, the quasiconformal mapping-based multiple-kernel model is solved with the constrained optimization. The framework maximizes the class discriminant ability of data in the nonlinear kernel-based manifold feature space. The kernel-based manifold learning system is improved.

2.2 Framework

In the section, we present a framework of kernel manifold learning with one example of kernel locality preserving projection (KLPP). KLPP preserves the local structure of the data in a low-dimensional mapping space. The objective function of KLPP is defined as

$$\min \sum_{i,j}^{n} \left\| z_i^{\Phi} - z_j^{\Phi} \right\|^2 S_{ij}^{\Phi} \tag{1}$$

where S_{ij}^{Φ} is a similarity matrix which measures the likelihood of two data points in the Hilbert space $\Phi(X) = [\Phi(x_1), \Phi(x_2), \ldots, \Phi(x_n)]$. The similarity matrix has the different meanings in the practical applications, for example, in hyperspectral image, and the similarity matrix describes the similarity of the different hyperspectral curves. $z_i^{\Phi} = (w^{\Phi})^T \Phi(x_i)$ is the low-dimensional representation of $\Phi(x_i)$ with the a projection vector w^{Φ}. On the similarity matrix S_{ij}^{Φ}, many methods are proposed to construct it in the previous work [11], and in this paper, we use the following formulation:

$$S_{ij}^{\Phi} = \begin{cases} \dfrac{k(x_i, x_j)}{\sqrt{k(x_i, x_i)}\sqrt{k(x_j, x_j)}} & \text{if } x_i \text{ and } x_j \text{ belong to the same class;} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Accordingly, Eq. (1) is changed to

$$\frac{1}{2} \sum_{i,j}^{n} \left\| z_i^{\Phi} - z_j^{\Phi} \right\|^2 S_{ij}^{\Phi} = \beta^T K \left(D^{\Phi} - S^{\Phi} \right) K \beta, \tag{3}$$

where K is kernel matrix calculated with the training samples, i.e., $K = Q^T Q$, and $D^{\Phi} = diag \left[\sum_j S_{1j}^{\Phi}, \sum_j S_{2j}^{\Phi}, \ldots, \sum_j S_{nj}^{\Phi} \right]$. The matrix D^{Φ} measures the importance of the data points. The element of the similarity matrix is larger, and the relationship of the data is more important. The relationship describes the manifold structure of two training samples. The multikernels-based quasiconformal kernel performs higher than the single quasiconformal kernel on the data distribution.

The hyperspectral image classification system classifies the curves from the different training with the similarity matrix. Accordingly, on kernel-based LPP, the constraint $(Z^{\Phi})^T D^{\Phi} Z^{\Phi} = 1$ can be rewritten as $\beta^T K D^{\Phi} K \beta = 1$. So, the minimization problem is transformed to

$$\begin{aligned} &\min_{\beta} \beta^T K L^{\Phi} K \beta \\ &\text{Subject to } \beta^T K D^{\Phi} K \beta = 1 \end{aligned} \tag{4}$$

where $L^{\Phi} = D^{\Phi} - S^{\Phi}$. QR decomposition of matrix K is considered as $K = P \Lambda P^T$, where $P = [r_1, r_2, \ldots, r_m]$, and $\Lambda = diag(\lambda_1, \lambda_2, \ldots, \lambda_m)$, and $r_1, r_2, \ldots, r_m$ are K's orthonormal eigenvectors corresponding to m largest nonzero eigenvalue $\lambda_1, \lambda_2, \ldots, \lambda_m$.

As above discussion, the previous work [11] proposed kernel locality preserve projections (KLPP) to improve locality preserve projections (LPP) on the nonlinear feature extraction. But KLPP still endures kernel model selection and parameters, and the performance is influenced by the kernel and parameters. Based on the basic framework of KLPP [11], we propose a manifold framework of multiple-kernel learning based on the quasiconformal mapping-based multiple-kernel model, and the parameter optimization method is proposed based on the optimization equation. The optimization objective equation is created to maximize the class discriminant ability of data in the nonlinear manifold feature space.

Secondly, as the excellent work [25], Lin presented supervised kernel-optimized LPP (SKOLPP) for face recognition and palm biometrics. The recognition performance is to maximize the class separability in kernel learning for feature extraction of image database. The excellent performances were reported on ORL, Yale, AR, and Palmprint databases. In this work, authors apply the data-dependent kernel to SKLPP, and authors claimed that the nonlinear features extracted by SKOLPP had larger discriminative ability compared with SKLPP. Lin's work testified the feasibility of enhancing the recognition performance with adjusting the kernel parameters of kernel model. So, the Lin's work aims to enhance the recognition performance of manifold learning, and we also aim to enhance the manifold learning recognition only with the different ideas as follows. SKOLPP applied the single-kernel method to supervised manifold learning, while our work applies multiple-kernel models. From the viewpoint of kernel optimization, quasiconformal multiple kernels have more discriminative ability than single-kernel learning.

So, the manifold framework of quasiconformal kernels learning is defined as

$$\begin{aligned} &\min_{\beta} \beta^T K^{(\alpha^*)} L^{\Phi} K^{(\alpha^*)} \beta \\ &\text{Subject to } \beta^T K^{(\alpha^*)} D^{\Phi} K^{(\alpha^*)} \beta = 1 \end{aligned} \tag{5}$$

where the optimal projection β is the main projection vector to construct the projection matrix, $K^{(\alpha^*)}$ is the kernel matrix with the optimal α^* of multikernels-based quasiconformal kernel $k(x, x') = K_f\left(k_{0,i}(x, x'), \mathbf{d}, \boldsymbol{\alpha}\right)$, so in this version, $\alpha^* = \{\mathbf{d}^*, \boldsymbol{\alpha}^*\}$, where $\mathbf{d}, \boldsymbol{\alpha}$ are adjusted for the classification task.

The optimal projection β^* is the manifold projection vector, and α^* is vector of the optimal kernel parameters. In the computing stages, we solve the α^* to obtain the optimal kernel matrix $\mathrm{K}^{(\alpha^*)}$ and then solve β^* the under the optimal kernel matrix $\mathrm{K}^{(\alpha^*)}$. The dimensions of the two vectors are determined by the practical applications. The optimal projection β^* will determine the feature vector after the dimensionality reduction of the data. And then the vector of optimal kernel parameters α^* is determined by the vector of expansion vector. The different number of the expansion vectors has the heavy influence on optimization performance of the learning system. The larger dimension increases the large computation stress. Accordingly, the computation efficiency is increased by the large-dimensional vectors.

2.3 Procedural steps

In this section, as the proposed framework of the kernel-based manifold learning, we solve the α^* to obtain the optimal kernel matrix $\mathrm{K}^{(\alpha^*)}$ and then solve β^* the under the optimal kernel matrix $\mathrm{K}^{(\alpha^*)}$. Accordingly, the procedure is described two steps: *Step 1. solving* $\alpha^* = \{\mathbf{d}^*, \boldsymbol{\alpha}^*\}$*; Step 2. solving the optimal projection* β^*. The detailed information is listed as follows.

Step 1. Solving $\alpha^* = \{\mathbf{d}^*, \boldsymbol{\alpha}^*\}$

Following the work in [11], we extend the quasiconformal kernel to quasiconformal multikernels. Different from the single quasiconformal kernel, only the expansion parameters need to be computed by constrained optimization equation. While on the quasiconformal multikernels, the weight parameter and expansion coefficient $\mathbf{d}, \boldsymbol{\alpha}$ are computed through the optimization equation, and the quasiconformal multikernels model has the higher ability on describing the data distribution than the quasiconformal kernel. According to the definition of the quasiconformal kernel [11], the geometrical structure of the data in the kernel mapping space is determined by the expansion coefficients with the determinative XVs and the free parameter. The structure is the data distributions in the empirical mapping space, and the kernel mapping space is empirical mapping space. The multikernels-based quasiconformal kernel has the higher ability on describing the data distribution than the quasiconformal kernel. The quasiconformal multikernels model is defined as

$$k(x,x') = f(x)\sum_{i=1}^{m} d_i k_{0,i}(x,x') f(x'), \tag{6}$$

where $k_{0,i}(x,x')$ is the i th basic kernel of polynomial kernel and Gaussian kernel, and m is the number of basic kernels for combination, $a_i \geq 0$ is the weight for the ith basic kernel function, $q(\cdot)$ is the factor function defined by $f(x) = \alpha_0 + \sum_{i=1}^{n} \alpha_i k_0(\mathbf{x}, a_i)$, where $k_0(x,a_i) = e^{-\gamma\|\mathbf{x}-a_i\|^2}$, $a_i \in R^d$, α_i is the coefficient for the combination, $\{a_i, i = 1,2,\ldots,n\}$ are selected by the training samples. The extended definition will not influence the characters of kernel matrix, and $k(x,x')$ satisfies the Mercer condition. Supposed that $\mathbf{d} = [d_1, d_2, \ldots, d_m]$, $\boldsymbol{\alpha} = [\alpha_0, \alpha_1, \alpha_2, \ldots, \alpha_n]$, the quasiconformal multikernels model is defined as

$$k(x,x') = K_f\left(k_{0,i}(x,x'), \mathbf{d}, \boldsymbol{\alpha}\right), \tag{7}$$

where $\mathbf{d}, \boldsymbol{\alpha}$ are the adjusted for the classification task. So, the jointly convex formulation can be described as

$$\begin{aligned} &\max_{\mathbf{d},\boldsymbol{\alpha}} F_c(K_0, \mathbf{d}, \boldsymbol{\alpha}) \\ &\text{Subject to } \|\mathbf{d}\| = 1, \|\boldsymbol{\alpha}\| = 1, \end{aligned} \tag{8}$$

$F_c(.)$ measures the class discriminative ability, and we can solve $\mathbf{d}, \boldsymbol{\alpha}$ with the two stages, one is to solve $\mathbf{d}$, and second is to solve $\boldsymbol{\alpha}$. In the first stage, the centered kernel alignment [26] is applied to create the objective optimization function, and in the second stage, Fisher-based and Margin-based optimization function is created to solve $\boldsymbol{\alpha}$.

Step 1.1. Optimize the weights of multiple kernels $\mathbf{d}$

The parameter vector $\mathbf{d} = [d_1, d_2, \ldots, d_m]$ is computed with centered kernel alignment [26] as follows.

$$\begin{aligned} &\max O_c(K_0^{(C)}, K^*) \\ &\text{Subject to } K_0^{(C)} = \sum_{i=1}^{m} d_i K_{0,i}^{(C)},\ tr(K_\mathbf{0}) = 1,\ d_i \geq 0,\ \forall i \end{aligned} \tag{9}$$

where $O_c(K_0^{(C)}, K^*)$ is the optimization objective function, and there are many methods to construct this equation, where $K^*(x,x') = \begin{cases} 1 & \text{if } y = y' \\ -1/(c-1) & \text{if } y \neq y' \end{cases}$ is the ideal target kernel, tr denotes the trace of a matrix. $K_0^{(C)} = [I - \frac{\mathbf{1}\mathbf{1}^T}{m}]K_0[I - \frac{\mathbf{1}\mathbf{1}^T}{m}]$ is the centered kernel matrix of K_0, I is the identity matrix, $\mathbf{1}$ is a vector with all entries equal to 1. Accordingly, $K_{0,i}^{(C)} = [I - \frac{\mathbf{1}\mathbf{1}^T}{m}]K_{0,i}[I - \frac{\mathbf{1}\mathbf{1}^T}{m}]$ is the centered kernel matrix of $K_{0,i}$, $i = 1,2,\ldots,m$. The objective function $O_c(K_0^{(C)}, K^*) = \frac{\langle K_0^{(C)}, K^* \rangle_F}{\|K_C^*\|_F \|K_0^{(C)}\|_F}$, K_C^* is the centered kernel matrix of K^*, where $\langle \cdot, \cdot \rangle_F$ is the Frobenius norm between two matrices, i.e., $\langle D, E \rangle_F = \sum_{i=1}^{m} \sum_{j=1}^{m} d_{ij} e_{ij} = tr(DE^T)$.$\|K_C^*\|_F$ is a unchanged value in the practical machine learning, while the trace constraint is to fix the scale invariance of KA. So, in the practical applications, we can consider the denominator $\|K_C^*\|_F$ as the unchanged value for the classification task. So, in the optimization equation of maximizing the numerator $\left\langle K_0^{(C)}, K^* \right\rangle_F$, and it can be removed during solving the equation.

Based on this, centered kernel alignment-based optimization problem can be transformed into a quadratic programming (QP) problem [27, 28], which is effectively solved with OPTI toolbox [26]. Supposed that $\mathbf{d} = [d_1, d_2, \ldots, d_m]$, the optimized solution $\mathbf{d}^*$ can be obtained through solving the following QP problem [26],

$$\begin{aligned} &\min v^T \mathbf{T} v - 2v^T \boldsymbol{\eta} \\ &\text{Subject to } vi \geq 0, \forall i \end{aligned} \tag{10}$$

where $\mathbf{d}^* = v^*/\|\mathbf{d}^*\|_2$, $\boldsymbol{\eta} = \left[\left\langle K_{0,1}^{(C)}, K^* \right\rangle_F \left\langle K_{0,2}^{(C)}, K^* \right\rangle_F, \ldots, \left\langle K_{0,m}^{(C)}, K^* \right\rangle_F\right]^T$, and $\mathbf{T}$ is a symmetric matrix defined by $\mathbf{T}_{ij} = \left\langle K_{0,i}^{(C)}, K_{0,j}^{(C)} \right\rangle_F, i,j = 1,2,\ldots,m.$

Step 1.2. Optimize the coefficients of quasiconformal kernel $\boldsymbol{\alpha}$

Fisher-based and margin-based optimization function is created to solve $\boldsymbol{\alpha}$. In the practical application, we can choose any method. The detail procedure is defined as follows.

Based on Fisher criterion [11], this step is to optimize the coefficients $\boldsymbol{\alpha} = [b_1, b_2, \ldots, b_n]$ of the quasiconformal multikernels based on Fisher criterion. Fisher criterion is to measure the class discriminative ability of the different class data, and this equation is increasing with the increasing of the class ability of data. The function is defined as

$$J_{Fisher}(\boldsymbol{\alpha}) = (\alpha^\mathbf{T} \mathbf{E}^\mathbf{T} \mathbf{B_0} \mathbf{E} \alpha)/(\alpha^\mathbf{T} \mathbf{E}^\mathbf{T} \mathbf{W_0} \mathbf{E} \alpha) \tag{11}$$

where $\mathbf{E}^\mathbf{T}\mathbf{B_0}\mathbf{E}$ and $\mathbf{E}^\mathbf{T}\mathbf{W_0}\mathbf{E}$ are constant matrices of training samples, and the objective function based on Fisher criterion is to measure the ability of the class discriminant. Then $\frac{\partial J_{Fisher}(\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}} = \frac{2}{\mathbf{J}_2^2}(\mathbf{J}_2 \mathbf{E}^\mathbf{T}\mathbf{B_0}\mathbf{E} - \mathbf{J}_1 \mathbf{E}^\mathbf{T}\mathbf{W_0}\mathbf{E})\boldsymbol{\alpha}$, where J_{Fisher} is solved through solving the eigenvalue problem of $(\mathbf{E}^\mathbf{T}\mathbf{W_0}\mathbf{E})^{-\mathbf{1}}(\mathbf{E}^\mathbf{T}\mathbf{B_0}\mathbf{E})$, and the eigenvector is the expansion coefficients $\boldsymbol{\alpha}$. However, in many applications, the matrix

$(\mathbf{E}^{\mathbf{T}}\mathbf{W_0}\mathbf{E})^{-\mathbf{1}}(\mathbf{E}^{\mathbf{T}}\mathbf{B_0}\mathbf{E})$ is not symmetrical, or the matrix $\mathbf{E}^{\mathbf{T}}\mathbf{W_0}\mathbf{E}$ is singular. Supposed that learning rate $\varepsilon(n) = \varepsilon_0(1-\frac{n}{N})$ of the initialized learning rate ε_0, the current iteration n, the total iterations N. The optimal $\boldsymbol{\alpha}$ is solved as

$$\boldsymbol{\alpha}^{(n+1)} = \boldsymbol{\alpha}^{(n)} + \varepsilon\left(\frac{1}{J_2}\mathbf{E}^{\mathbf{T}}\mathbf{B_0}\mathbf{E} - \frac{J_{Fisher}}{J_2}\mathbf{E}^{\mathbf{T}}\mathbf{W_0}\mathbf{E}\right)\boldsymbol{\alpha}^{(n)} \tag{12}$$

Based on *maximum margin criterion* [11], the objective function is defined as

$$\begin{aligned}&\max\ \boldsymbol{\alpha}^{\mathbf{T}}\left(2\tilde{\mathbf{S}}_{\mathbf{B}} - \tilde{\mathbf{S}}_{\mathbf{T}}\right)\boldsymbol{\alpha}\\ &\text{Subject to } \boldsymbol{\alpha}^{\mathbf{T}}\boldsymbol{\alpha} - 1 = 0\end{aligned} \tag{13}$$

The optimal expansion coefficient $\boldsymbol{\alpha}^*$ is the eigenvector of $2\tilde{\mathbf{S}}_{\mathbf{B}} - \tilde{\mathbf{S}}_{\mathbf{T}}$. $\begin{cases}\tilde{\mathbf{S}}_{\mathbf{B}} = \mathbf{X_B}\mathbf{X}_{\mathbf{B}}^{\mathbf{T}}\\ \tilde{\mathbf{S}}_{\mathbf{T}} = \mathbf{X_T}\mathbf{X}_{\mathbf{T}}^{\mathbf{T}}\end{cases}$, $\begin{cases}\mathbf{X_T} = (\mathbf{Y}_0 - \frac{1}{\mathbf{m}}\mathbf{Y}_0\mathbf{1}_{\mathrm{m}}^{\mathbf{T}}\mathbf{1_m})\mathbf{E}\\ \mathbf{X_B} = \mathbf{Y_0}\mathbf{M}^{\mathbf{T}}\mathbf{E}\end{cases}$, $\mathbf{M} = \mathbf{M}_1 - \mathbf{M}_2$ and $\mathbf{M}_1$, $\mathbf{M}_2$ are defined as

$$M_1 = \begin{bmatrix} \left[\frac{1}{\sqrt{m_1}}\right]_{m_1\times m_1} & 0_{m_1\times m_2} & \cdots & 0_{m_1\times m_c}\\ 0_{m_2\times m_1} & \left[\frac{1}{\sqrt{m_2}}\right]_{m_2\times m_2} & \cdots & 0_{m_2\times m_c}\\ \vdots & \vdots & \ddots & \vdots\\ 0_{m_c\times m_1} & 0_{m_c\times m_2} & \cdots & \left[\frac{1}{\sqrt{m_c}}\right]0_{m_c\times m_c} \end{bmatrix} \text{ and}$$

$$M_2 = \begin{bmatrix} \frac{\sum_j^c\sqrt{m_j}}{m} & 0 & \cdots & 0\\ 0 & \frac{\sum_j^c\sqrt{m_j}}{m} & & \\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \frac{\sum_j^c\sqrt{m_j}}{m} \end{bmatrix}.$$

$\mathbf{Y}_0 = \mathbf{K_0}\mathbf{P_0}\boldsymbol{\Lambda}_0^{-1/2}$, $\mathbf{K}_0 = \mathbf{P_0}\boldsymbol{\Lambda}_0^{\mathbf{T}}\mathbf{Y}_0^{\mathbf{T}}$, and K_0 is the basic matrix. So

$$\begin{cases}\mathrm{trace}(\mathrm{SB}) = \boldsymbol{\alpha}^{\mathbf{T}}\mathbf{X}_{\mathbf{B}}^{\mathbf{T}}\mathbf{X_B}\boldsymbol{\alpha}\\ \mathrm{trace}(\mathrm{S_T}) = \boldsymbol{\alpha}^{\mathbf{T}}(\mathbf{X}^{\mathbf{T}})^{\mathbf{T}}\mathbf{X_T}\boldsymbol{\alpha}\end{cases} \tag{14}$$

Supposed that $\tilde{\mathbf{S}}_{\mathbf{B}} = \mathbf{X_B}\mathbf{X}_{\mathbf{B}}^{\mathbf{T}}$ and $\tilde{\mathbf{S}}_{\mathbf{T}} = \mathbf{X_T}\mathbf{X}_{\mathbf{T}}^{\mathbf{T}}$ and then

$$Dis(\boldsymbol{\alpha}) = \mathbf{trace}\left(\boldsymbol{\alpha}^{\mathbf{T}}\left(2\tilde{\mathbf{S}}_{\mathbf{B}} - \tilde{\mathbf{S}}_{\mathbf{T}}\right)\boldsymbol{\alpha}\right) \tag{15}$$

So, maximizing $Dis(\boldsymbol{\alpha})$ is equal to obtain the objective function through calculating eigenvalue equation of matrix $2\tilde{\mathbf{S}}_{\mathbf{B}} - \tilde{\mathbf{S}}_{\mathbf{T}}$, the column vector of P is the eigenvalue matrix of $2\tilde{\mathbf{S}}_{\mathbf{B}} - \tilde{\mathbf{S}}_{\mathbf{T}}$, the eigenvalue is $\mathbf{2\Lambda} - \mathbf{I}$.

Step 2. solving the optimal projection β^*

After computing, $K^{(\alpha^*)}$ is computed by the kernel matrix with the optimal α^* of multikernels-based quasiconformal kernel $k(x,x') = K_f\left(k_{0,i}(x,x'),\mathbf{d},\boldsymbol{\alpha}\right)$. Then we solve the manifold optimization (5), $\min\limits_{\beta}\beta^T K^{(\alpha^*)}L^{\Phi}K^{(\alpha^*)}\beta$, subject to $\beta^T K^{(\alpha^*)}D^{\Phi}K^{(\alpha^*)}\beta = 1$, where the optimal projection β is the main projection vector to construct the projection matrix. QR decomposition of matrix $K^{(\alpha^*)}$ is considered as $K^{(\alpha^*)} = P\Lambda P^T$, where $P = [r_1, r_2, \ldots, r_m]$, and $\Lambda = diag(\lambda_1, \lambda_2, \ldots, \lambda_m)$, and $r_1, r_2, \ldots, r_m$ are $K^{(\alpha^*)}$'s orthonormal eigenvector corresponding to m largest nonzero eigenvalue $\lambda_1, \lambda_2, \ldots, \lambda_m$.

2.4 Procedural flowchart

The procedure is shown in Fig. 1. The procedure includes three procedures of multiple kernels optimization, training and testing for general kernel-based manifold learning application. This proposed procedure costs more time to solve the optimization equation than KLPP without optimizing kernel function. But the kernel optimization procedure can be implemented off-line. In the practical applications, the kernel optimization procedure is implemented offline, and in the application it needs the additional less time consumption. So, it has the little influence on the learning efficiency in the online application. The optimization equation is solved by iteration method and eigenvalue decomposition method, and they are the most popular methods of solving the optimization equation. Other methods also are the one of the two kinds of optimization method.

The advance beyond the state-of-the-art comes from the two points. (1) A manifold framework of quasiconformal multikernels learning is proposed for hyperspectral data on dimensionality reduction. Compared to the traditional kernel-based manifold learning, the proposed quasiconformal multikernels manifold learning achieves the class discriminant ability of data for the data classification. (2) The quasiconformal mapping-based multiple-kernel model is proposed for kernel mapping, and the model can adaptively change the kernel mapping structure of data distribution. The proposed kernel-based manifold learning has the more ability of describing the data mapping than the traditional kernel and data-dependent kernel model, because the data distribution structure can be adaptively adjusted.

2.5 Discussion

The proposed kernel-based manifold learning is based on the kernel-based machine. The theoretical bounds also come from quasiconformal multikernels. The application system is also to optimize accuracy in predicting the test

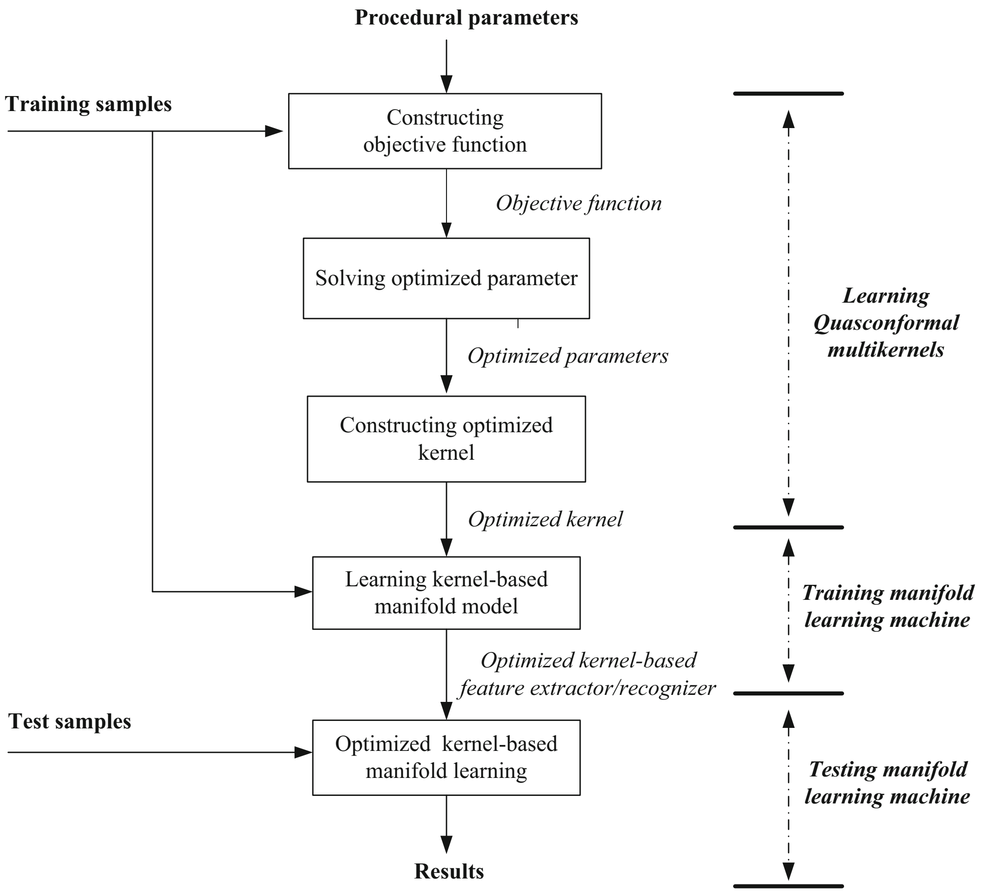
Procedural parameters
Training samples
Constructing objective function
Objective function
Solving optimized parameter
Optimized parameters
Constructing optimized kernel
Optimized kernel
Learning kernel-based manifold model
Optimized kernel-based feature extractor/recognizer
Test samples
Optimized kernel-based manifold learning
Results
Learning Quasconformal multikernels
Training manifold learning machine
Testing manifold learning machine

Fig. 1 Procedure of multiple kernels-based manifold learning

data based on train test. Supposed that the training and test sets have the same size of data set, we can show a performance guarantee that holds with high probability over uniformly chosen training/test partitions.

For a function $f : \chi \rightarrow R$, the proportion of errors on the test data of a threshold version of f can be written as

$$er(f) = \frac{1}{n}|\{n+1 \le i \le 2n : y_i f(x_i) \le 0\}| \tag{16}$$

where the kernel classifiers were obtained by thresholding kernel expansions of the form, with the bounded norm,

$$\|w\|^2 = \sum_{i,j=1}^{2n} \alpha_i \alpha_j k(x_i, x_j) = \boldsymbol{\alpha}^T K \boldsymbol{\alpha} \le \frac{1}{\gamma^2} \tag{17}$$

where K is the quasiconformal multikernel.

It also holds with high probability over the choice of the training and test data because permuting the sample leaves the distribution unchanged. Here we provide an upper bound on the error of a kernel classifier on the test data over the training data of a certain margin cost function with properties of the kernel matrix. In this paper, we focus on the 1-norm soft margin classifier after manifold-based feature extraction [26].

For every $\gamma > 0$ with probability at least $1 - \delta$ over the data (x_i, y_i), every function $f \in F_k$ has $er(f)$ no more than

$$\frac{1}{n}\sum_{i=1}^{n} \max(1 - y_i f(x_i)) + \frac{1}{\sqrt{n}}\left(4 + \sqrt{2\log(1/\delta)} + \sqrt{\frac{\varsigma(k)}{n\gamma^2}}\right) \tag{18}$$

where $\varsigma(\mathrm{K}) = E \max_{K \in \mathrm{K}} \sigma^T K \sigma$ with the expectation over σ. So,

$$\varsigma(\mathrm{K}) = cE \max_{K \in \mathrm{K}} \sigma^T \frac{K}{trace(K)} \sigma \tag{19}$$

then

$$\varsigma(\mathrm{K}) \le c\min\left(m, n \max_{j} \frac{\lambda_j}{trace(K_j)}\right) \tag{20}$$

where λ_j is the largest eigenvalue of K_j. So, the test error is bounded by a sum of the average over the training data of a margin cost function plus a complexity penalty term that depends on the ratio between the trace of the quasiconformal multikernel kernel matrix and the squared margin parameter, γ^2.

3 Experimental results

3.1 Experimental and procedural parameters setting

We evaluate the performances on the databases, and the recognition accuracy is evaluated as the performance index. The average recognition accuracy of ten times of experiments is used to evaluate the classification performance. The experiments are implemented in the MATLAB platform (Version 6.5) with the computer of Pentium 3.0 GHz, 512 MB RAM. On the selection of the procedural parameters, the cross-validation method is applied to select the procedure parameters. We choose the kind of basic kernel functions for the different application systems, and the parameter of basic kernel is chosen with cross-validation method.

3.2 Performance on ORL and Yale databases

In this section, some experiments on YALE and ORL databases were implemented to evaluate the unified framework of multiple kernels manifold learning. YALE database from the YALE Center for Computational Vision and Control contains 165 grayscale images of 15 individuals, and ORL database was developed at the Olivetti Research Laboratory that consisted of 400 images from 40 individuals. The original ORL face images of 112 × 92 pixels are resized to 48 × 48 pixels. On the Yale database, we divide the database into 5 sub-databases through selecting randomly 5 samples as the training set, and the rest samples as the test sample. The sub-datasets are denoted with T1, T2, T3, T4, and T5. We implement LPP [11], CLPP [11], KCLPP [11], and our method to evaluate the performance, and the experimental results are shown in Table 1, and the procedural parameters were chosen through the cross-validation. Similarly, on ORL database, the quasiconformal kernel-based manifold learning methods are comprehensively evaluated compared with LPP, CLPP, and KCLPP. As shown in Table 2, the proposed framework achieves the highest recognition accuracy because the data structure is adaptively changed for the input data.

Table 1 Recognition performance on Yale databases (%)

Datasets	LPP [11]	CLPP [11]	KCLPP [11]	Our method
T1	86.33	90.00	94.44	95.67
T2	90.67	91.11	92.22	93.33
T3	88.56	86.67	93.33	94.44
T4	88.89	90.00	93.33	92.33
T5	95.56	93.33	96.67	97.44

Table 2 Recognition performance on ORL databases (%)

Datasets	LPP [11]	CLPP [11]	KCLPP [11]	Our method
T1	95.25	96.25	96.25	98.25
T2	93.75	94.25	95.25	97.25
T3	95.25	97.50	98.25	99.25
T4	93.50	94.25	96.25	97.25
T5	91.25	92.25	96.25	97.00

Table 3 Recognition performance on ORL and Yale databases on randomly selection (%)

Datasets	LPP [11]	CLPP [11]	KCLPP [11]	Our method
ORL	94.55	95.35	95.75	98.15
YALE	86.33	91.23	93.44	95.67

Some evaluations are implemented on the randomly selection, and the experimental result is shown in Table 3. The 10 times of experiments are implemented, and averaged recognition accuracy is considered as the index of performance. As experimental results, the proposed kernel optimization method performs better than other methods.

3.3 Application to hyperspectral image classification

The framework of hyperspectral image classification system is shown in Fig. 2. Hyperspectral imagery is the most popular remote sensing technology on satellite platform, with the prospective applications in military monitoring, energy exploration, geographic information, and so on. The development of hyperspectral instruments with hundreds of contiguous spectral channels promotes collecting remote imagery data. The size of the data is largely increased with the high resolutions of spectral and space. Two problems occur in the practical applications: (1) the bandwidth of the communication channel limits the transmission of the full hyperspectral image data for the further processing and analysis on the ground; (2) the demand of the real-time processing for some applications. Data compression is feasible to solve the transmission problem, but is still endure the limitation on real-time analysis. So, based on the real-time image analysis, machine learning-based data analysis technology is feasible and effective to produce one image from the full band of hyperspectral images. The classification is to classify the spectrum curve based on the spectrum data of each object. The hyperspectral data machine learning system is implemented on the satellite platform. After the hyperspectral data collection, each pixel is classified and denoted to the different objects based on the spectrum database. The spectrum data in database are collected in advance, so it has inconsistency between the

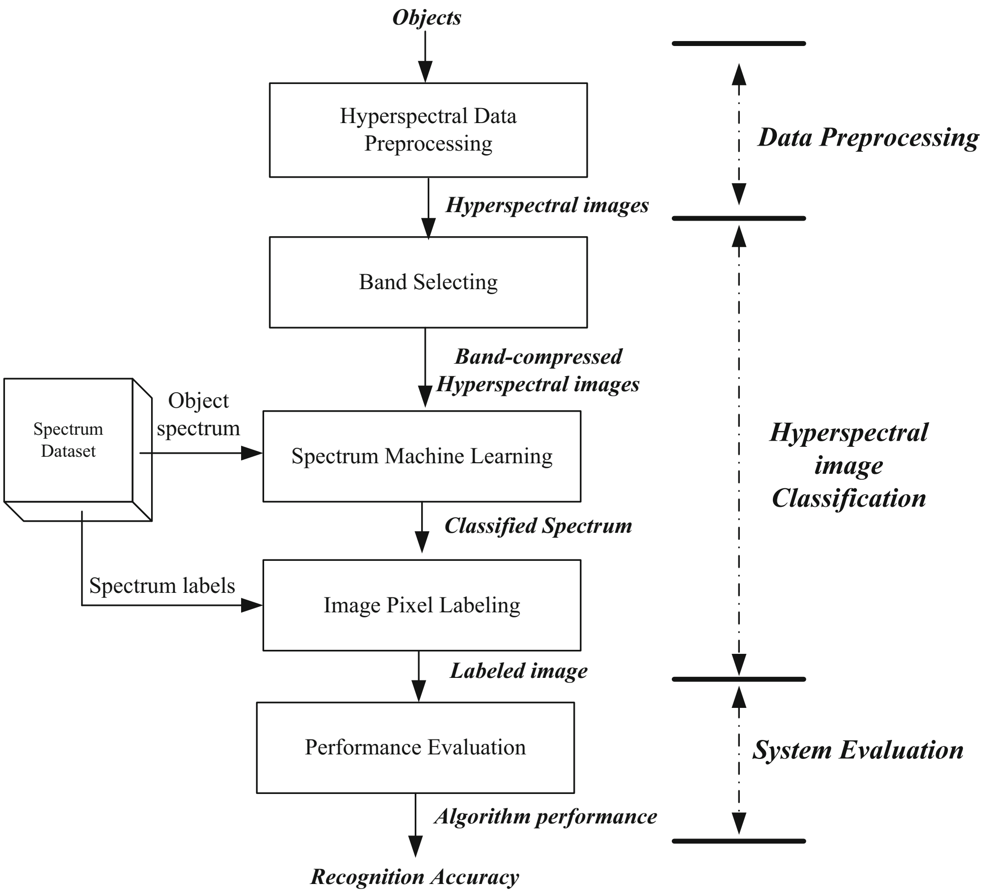

Fig. 2 Application framework of kernel-based manifold learning

spectrums with the data collection. The inconsistency can be considered the nonlinear changing. The relationship between spectral curves is the classical nonlinear relationship. So the classification is the nonlinear and complex classification problem. Researches show that kernel learning method is not effective to hyperspectral sensing data. Kernel-based manifold learning is applied to hyperspectral sensing data classification.

Based on the application framework, we evaluate the proposed algorithm on Indian Pines and Washington, D.C. Mall databases. The two databases have the various spectral and spatial resolutions under the different environments of remote sensing.

Indian Pines dataset is collected based on airborne platform on June 1992 and has the various spectral and spatial resolutions, and the spectral curves denote the different remote sensing environments. The airborne visible/infrared imaging spectrometer (AVIRIS) data cube has 224 bands of spectral resolution, and it has the spatial resolution of 20 m per pixel. In our experiments, we removed the noisy and water–vapor absorption bands and 200 bands of images are used in the experiments. The whole scene is consists of 145 × 145 pixels, and 16 classes of interested objects rang the size from 20 to 2468 pixels, but only 9 classes of objects are selected in the experiments. Some examples are shown in Fig. 3.

D.C. Mall data were acquired under the airborne with hyperspectral digital imagery collection experiment (HYDICE) sensor on August 23, 1995. The image has 1280 × 307 pixels, and it has the spatial resolution of 1.5 m, and 210 spectral bands are in the 0.4–2.4-μm region. In the experiments, several bands influenced by the atmospheric absorption are ignored, and the rest 191 bands are implemented in the experiments. The image is resized to the size of 211 × 307 including 7 classes of land-covers namely roof, grass, street, trees, water, path and shadow. Some examples are shown in Fig. 4.

Firstly, we evaluate the proposed algorithm compared with support vector classifier (SVC), kernel sparse representation classifier (KSRC) on data classification. On the basic kernel functions, we compare them in the practical hyperspectral image classification. We test the single-kernel and quasiconformal multikernels for kernel classifiers on SVC and KSRC, that is, PK-SVC: polynomial kernel-SVC, GK-SVC: Gaussian kernel-SVC, QMK-SVC: quasiconformal multikernels-based SVC, PK-KSRC:

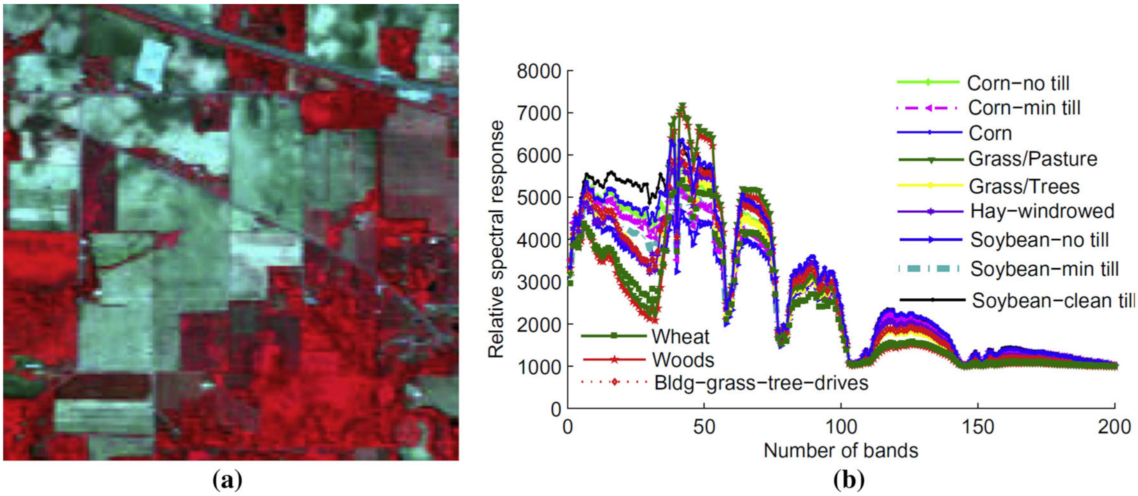

Fig. 3 One example of Indian Pines data in He & Li [29]. **a** Three band false color composite, **b** spectral signatures

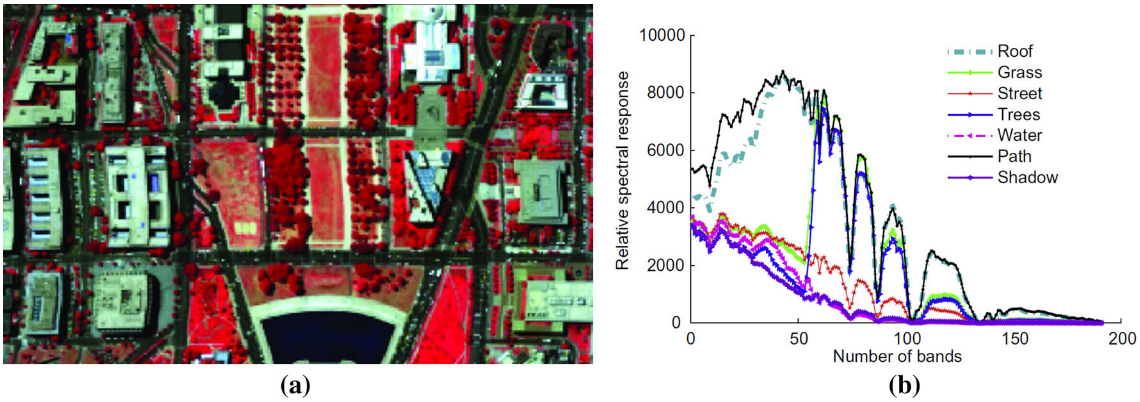

Fig. 4 One example from D.C. Mall data in He & Li [29]. **a** Three band false color composite, **b** spectral signatures

polynomial kernel-KSRC, GK-KSRC: Gaussian kernel-KSRC, QMK-KSRC: quasiconformal multikernels-based KSRC. For the quantitative comparison, we implement some experiments using polynomial kernel-KCLPP (PK-KCLPP), Guassian kernel (GK-KCLPP), multiple kernel (MK-KCLPP), and quasiconformal multiple kernel-based KCLPP (QMK-KCLPP), PK-SVC, GK-SVC, QMK-SVC, PK-KSRC, GK-KSRC, and QMK-KSRC. The averaged accuracy is to evaluate the performance of the algorithms, and the experimental results are shown in Tables 4 and 5. On the SVC, QMK-SVC performs better than PK-SVC and GK-SVC. On the KSRC, QMK-KSRC outperforms PK-KSRC and GK-KSRC. In particular, the polynomial kernel performs better than Gaussian kernel under SVC and KSRC classifiers. On selection of the basic kernels for multiple-kernel learning, we select the Gaussian kernel and polynomial kernel as the basic kernels.

Moreover, we also implement some experiments on D.C. Mall data to evaluate the proposed framework including polynomial kernel-KCLPP (PK-KCLPP), Guassian kernel (GK-KCLPP), multiple kernel (MK-KCLPP), and quasiconformal multiple kernel-based KCLPP (QMK-KCLPP). The experimental results are shown in Table 6. The experimental results show that multiple kernels-based manifold learning performs better than the basic kernels, and quasiconformal multiple kernel-based manifold outperform learning performance better than multiple version. So, it is feasible to apply quasiconformal kernel model to improve the kernel manifold learning.

3.4 Discussion

As *experimental results on the performance* of manifold learning and other machine learning based on quasiconformal kernel and quasiconformal multiple kernels, we can conclude that, the quasiconformal kernel-based manifold learning performs better than basic kernel-based manifold learning, and quasiconformal multiple kernels outperform

Table 4 Performance of manifold learning compared with SVC on the Indian Pines data (%)

Class	1	2	3	4	5	6	7	8	9	10	11	12
PK-SVC	49.3	58.7	96.4	39.2	65.8	93.6	62.9	85.3	100	65.8	72.3	58.4
GK-SVC	78.0	73.6	99.1	76.9	80.5	97.1	79.7	89.8	99.7	83.6	86.0	80.7
QMK-SVC	78.3	80.4	99.9	82.5	90.2	99.2	82.7	98.5	100	86.8	90.2	84.4
KCLPP	76.2	71.4	97.3	74.1	77.4	95.5	76.3	87.1	96.5	81.1	84.6	78.8
QMK-KCLPP	79.3	81.1	98.9	83.2	89.9	98.5	84.7	98.8	100	88.7	90.6	84.8

Table 5 Performance of manifold learning compared with KSRC on the Indian Pines data (%)

Class	1	2	3	4	5	6	7	8	9	10	11	12
PK-KSRC	51.8	59.6	96.1	49.1	78.5	93.8	62.8	84.7	100	67.5	75.2	60.7
GK-KSRC	77.8	76.4	99.1	75.5	79.0	97.4	82.7	88.7	100	83.9	86.3	81.1
QMK-KSRC	79.4	83.5	99.8	83.4	92.6	99.4	82.8	98.3	100	87.8	91.0	85.6
KCLPP	78.9	77.5	99.1	76.5	79.0	98.4	83.7	89.7	100	84.9	87.3	82.1
QMK-KCLPP	79.9	83.7	99.8	83.7	92.9	99.9	83.3	99.1	100	88.8	92.0	86.6

Table 6 Performance on the D.C. Mall data (%)

Class	PK-KCLPP	GK-KCLPP	MK-KCLPP	QMK-KCLPP
1	76.23	84.83	90.38	91.12
2	95.45	94.27	96.62	97.45
3	90.24	94.35	95.86	96.67
4	94.45	95.22	95.57	96.35
5	99.67	99.39	97.49	98.25
6	99.22	99.23	97.93	98.27
7	93.23	94.28	95.64	96.83

other methods. Kernel trick is an effective method to solve the nonlinear problems of machine learning, and the recognition accuracy and prediction accuracy are largely increased with the nonlinear kernel mapping. There is no any kernel which is adaptive to all applications of detecting intrinsic information for the complicate sample data. The multiple kernel-based manifold learning has different kernel representations for the different feature subspaces, and multiple-kernel learning is a feature extraction method of combining many features. So, the multikernel-based manifold learning performs better than the single feature extraction on the data. The proposed framework is to solve the selection of function and parameter of kernel, which have heavy influences on the performance of kernel-based learning system. Quasiconformal single-kernel structure changes the data structure in the kernel empirical space. And then, quasiconformal multiple kernels are combined to more precisely characterize the data for improving performance on solving complex visual learning tasks, so the proposed framework outperforms others in the different datasets.

Moreover, we implement many experiments with the recognition accuracy, and we do not consider the efficiency in the experiments. The procedure includes three procedures of multiple kernels optimization, training and testing for general kernel-based manifold learning application. The optimization procedure costs more time, but the kernel optimization procedure can be implemented off-line. So the kernel optimization-based manifold learning does not cost much time on the online application. In the training steps, the optimal parameters are solved through iteration optimization, and the procedure will cost much time. While in the test stage, it needs the additional less time consuming. So, it has the little influence on the learning efficiency.

4 Conclusion

This paper presents a novel framework of manifold multiple-kernel learning, and it applies quasiconformal multiple-kernel model to increase the data description ability. Some experiments are implemented to evaluate the multiple kernel and quasiconformal kernel. The proposed framework performs better compared with the traditional methods. The framework preserves the good structure of data distribution for classification with the quasiconformal mapping-based multiple-kernel model. And the model has the maximum class discriminant ability of data in the nonlinear manifold feature space. So, the proposed method is a promising dimensionality reduction method on data processing, especially on hyperspectral image processing. The proposed manifold learning is a promising dimensionality reduction method on data processing, especially on hyperspectral data processing, and it preserves the local embedding. The proposed framework can be applied to many applications, for example, image retrieval, video classification, speech recognition, and so on.

References

1. Belhumeur PN, Hespanha JP, Kriegman DJ (1997) Eigenfaces vs. fisherfaces: recognition using class specific linear projection. IEEE Trans Pattern Anal Mach Intell 19(7):711–720
2. Batur AU, Hayes MH (2001) Linear subspace for illumination robust face recognition. In: Proceedings of the IEEE international conference on computer vision and pattern recognition, pp 296–301
3. Hastie T, Stuetzle W (1989) Principal curves. J Am Stat Assoc 84:502–516
4. Chang K-Y, Ghosh J (2001) A unified model for probabilistic principal surfaces. IEEE Trans Pattern Anal Mach Intell 23(1):22–41
5. Zhu Z, He H, Starzyk JA, Tseng C (2007) Self-organizing learning array and its application to economic and financial problems. Inf Sci 177(5):1180–1192
6. Yin H (2002) Data visualisation and manifold mapping using the ViSOM. Neural Netw 15(8):1005–1016
7. Roweis ST, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding. Science 290:2323–2326
8. Tenenbaum JB, de Silva V, Langford JC (2000) A global geometric framework for nonlinear dimensionality reduction. Science 290:2319–2323
9. He X, Niyogi P (2003) Locality preserving projections. In: Proceedings of the conference on advances in neural information processing systems, pp 585–591
10. He X, Yan S, Hu Y, Niyogi P, Zhang H (2005) Face recognition using Laplacianfaces. IEEE Trans Pattern Anal Mach Intell 27(3):328–340
11. Li J-B, Pan J-S, Chu S-C (2008) Kernel class-wise locality preserving projection. Inf Sci 178(7):1825–1835
12. Mulier F, Cherkassky V (1995) Self-organization as an iterative kernel smoothing process. Neural Comput 7:1165–1177
13. Ritter H, Martinetz T, Schulten K (1992) Neural computation and self-organizing maps. Addison-Wesley, Reading, pp 64–72
14. Chen C, Li W, Hongjun S, Liu K (2014) Spectral–spatial classification of hyperspectral image based on kernel extreme learning machine. Remote Sens 6(6):5795–5814
15. Huang J, Yuen PC, Chen W-S, Lai JH (2004) Kernel subspace LDA with optimized kernel parameters on face recognition. In: Proceedings of the sixth IEEE international conference on automatic face and gesture recognition
16. Pan JS, Li JB, Lu ZM (2008) Adaptive quasiconformal kernel discriminant analysis. Neurocomputing 71(13–15):2754–2760
17. Chen W-S, Yuen PC, Huang J, Dai D-Q (2005) Kernel machine-based one-parameter regularized fisher discriminant method for face recognition. IEEE Trans Syst Man Cybern B Cybern 35(4):658–669
18. Xiong H, Swamy MN, Ahmad MO (2005) Optimizing the kernel in the empirical feature space. IEEE Trans Neural Netw 16(2):460–474
19. Amari S, Wu S (1999) Improving support vector machine classifiers by modifying kernel functions. Neural Netw 12(6):783–789
20. Li J-B, Pan J-S, Lu Z-M (2009) Kernel optimization-based discriminant analysis for face recognition. Neural Comput Appl 18(6):603–612
21. Xie X, Li B, Chai X (2015) Kernel-based nonparametric fisher classifier for hyperspectral remote sensing imagery. J Inf Hiding Multimed Signal Process 6(3):591–599
22. Subrahmanya N, Shin YC (2010) Sparse multiple kernel learning for signal processing applications. IEEE Trans Pattern Anal Mach Intell 32(5):788–798
23. Sonnenburg S, Rätsch G, Schäfer C, Schölkopf B (2006) Large scale multiple kernel learning. J Mach Learn Res 7:1531–1565
24. Kloft M, Brefeld U, Sonnenburg S, Zien A (2011) lp-Norm multiple kernel learning. J Mach Learn Res 12:953–997
25. Lin C, Jiang J, Zhao X, Pang M, Ma Y (2015) Supervised kernel optimized locality preserving projection with its application to face recognition and palm biometrics. Math Probl Eng. doi:10.1155/2015/421671
26. Koltchinskii V, Panchenko D (2002) Empirical margin distributions and bounding the generalization error of combined classifiers. Ann Stat 30(1):1–50
27. Wang L, Chan KL, Xue P (2005) A criterion for optimizing kernel parameters in KBDA for image retrieval. IEEE Trans Syst Man Cybern B Cybern 35(3):556–562
28. Cortes C, Mohri M, Rostamizadeh A (2012) Algorithms for learning kernels based on centered alignment. J Mach Learn Res 13(1):795–828
29. He Z, Li J (2015). Multiple data-dependent kernel for classification of hyperspectral images. Expert Syst Appl 42(3):1118–1135