A Locality Preserving Approach for Kernel PCA

Yin Zheng[1](✉), Bin Shen[2], Xiaofeng Yang[1], Wanli Ma[1], Bao-Di Liu[3], and Yu-Jin Zhang[1]

[1] Department of Electronic Engineering, Tsinghua University, Beijing 10084, China
y-zheng09@mails.tsinghua.edu.cn
[2] Google Research, New York, USA
[3] College of Information and Control Engineering, China University of Petroleum, Qingdao 266580, China

Abstract. Dimensionality reduction is widely used in image understanding and machine learning tasks. Among these dimensionality reduction methods such as LLE, Isomap, etc., PCA is a powerful and efficient approach to obtain the linear low dimensional space embedded in the original high dimensional space. Furthermore, Kernel PCA (KPCA) is proposed to capture the nonlinear structure of the data in the projected space using "*Kernel Trick*". However, KPCA fails to consider the locality preserving constraint which requires the neighboring points nearer in the reduced space. The locality constraint is natural and reasonable and thus can be incorporated into KPCA to improve the performance. In this paper, a novel method, which is called Locality Preserving Kernel PCA (LPKPCA) is proposed to reduce the reconstruction error and preserve the neighborhood relationship simultaneously. We formulate the objective function and solve it mathematically to derive the analytical solution. Several datasets have been used to compare the performance of KPCA and our novel LPKPCA including ORL face dataset, Yale Face Dataset B and Scene 15 Dataset. All the experimental results show that our method can achieve better performance on these datasets.

Keywords: Locality preserving constraint · Kernel PCA · Dimensionality reduction

1 Introduction

Many problems in image understanding involve some kind of dimensionality reduction [1–6]. Recently, a lot of dimensionality reduction methods have been proposed such as PCA, LDA, LPP [1], Isomap [7], LLE [8], etc. Among all these methods, PCA is a powerful and popular linear technique to extract lower manifold structure from high dimensional data, which has been widely used in pattern recognition such as face recognition, object recognition, etc. PCA seeks the

This work was partially supported by the National Nature Science Foundation of China (NNSF: 61171118).

B. Shen—This work was done when Bin was with Department of Computer Science, Purdue University, West Lafayette.

Y.-J. Zhang (Ed.): ICIG 2015, Part I, LNCS 9217, pp. 121–135, 2015.
DOI: 10.1007/978-3-319-21978-3_12

optimal combination of the input coordinates which reduces the reconstruction error of the input data to form a low dimensional subspace. The corresponding new coordinates are called Principal Vectors. It is often the case that only a small number of the important Principal Vectors is good enough to represent the original data and can furthermore reduce the noise that is induced by the unimportant Principal Vector. PCA provides an efficient way to compress the data with minimal information loss using the eigenvalue decomposition of the data covariance matrix. In fact, the principal vectors are uncorrelated and form the closest linear subspace to the data, which is useful in subsequent statistical analysis.

A lot of variant PCA have been proposed to modify the performance of PCA. Alexandre and Aspremont [9] proposed DSPCA which was based on relaxing a hard cardinality cap constraint with a convex approximation. In [10], Ron Zass and Amnon Shashua proposed a nonnegtive sparse PCA to capture the nonnegtive and sparseness nature of the real world. What's more, as the real world observations are often corrupted by noise, the principal vectors might not be the ones we desired. Hence, people tried to make some efforts to make PCA be robust to the noisy observations [11–13]. For examples, Candes [11] proposed to decompose the observations into a low rank matrix and a noise item, which would make the model be robust to corruptions. And Goes [13] proposed three stochastic approximation algorithms for robust PCA which have smaller storage requirements and lower runtime complexity. Because PCA and its variants are linear transformation of their original space, they cannot capture the nonlinear structure of the data. However, some kind of data lies in the nonlinear structure subspace [1]. To solve this problem, Bernhard Scholkopf *et al.* [14,15] proposed a nonlinear form of PCA using kernel method, which was called Kernal PCA (KPCA). KPCA maps the original feature space into a high dimensional feature space and seeks the principle vectors in the mapped space. It uses "*Kernel Trick*" to solve the problem. It has been proved that KPCA outperforms PCA in pattern recognition problems with same number of principle vectors and the performance can be furthermore improved using more components than PCA. However, KPCA is suffered from the memory problem and computational efficiency problem in the situation when the number of training sample is large [16]. To solve the problem, M. Tippings [17] proposed to select a subset of the training samples to approximate the covariance matrix using a maximum likelihood approach. And Sanparith Marukatat also discovered the problem and proposed to use kernel K-means and preimage reconstruction algorithms to solve the problem. What's more, Honeine [18] proposed an online version of Kernel PCA to deal with large scale dataset. Another drawback for KPCA is that it fails to consider the intrinsic geometric structure of the data. The only objective function of KPCA and PCA is to reduce the reconstruction error of the data without considering the neighborhood relationship preserving constraint. But it is a natural and reasonable assumption that a good projection should map two data points which are close to each other in the original space into two points also close in the projected feature space [1–4]. However, KPCA does not take this constraint into consideration explicitly.

In this paper, we aim to solve the problem of KPCA which does not consider the intrinsic geometric structure of the data and propose a novel kernel PCA which preserves the locality constraint relationship in the original feature space using the graph of Laplacian [1,19] which incorporates the neighborhood relationship of the data. We call this novel method Locality Preserving Kernel PCA (LPKPCA).

This paper is organized as follows. The related works about locality preserving constraint are introduced in Sect. 2. Then a brief review about PCA and KPCA is given in Sect. 3. In Sect. 4 the new objective function and the derivation for LPKPCA are illustrated in details. In Sect. 5 the experiment results are shown to compare the performance between KPCA and LPKPCA on several datasets including ORL face dataset, Yale Face Dataset B and Scene 15 Dataset. Finally, in Sect. 6 a conclusion is given to summarize this paper and point out the future work.

2 Related Works

The concept of locality preserving dimensionality reduction can be traced back to [7,8]. The locality constraint requires the dimensionality reduction projection to preserve the neighborhood relationship, which has been proved to be a very reasonable assumption [1,20,21]. In [19], Mikhail Belkin and Partha Niyogi proposed to use laplacian eigenmap to find the low dimensional embedding of the data which lies in the original high dimensional feature space. X. He, *et al.* [1] extended this conception and proposed Locality Preserving Projection (LPP) to find the optimal linear approximation of to the eigenfunctions to the Laplace Beltrami operator on the manifold. Although LPP is a linear transformation, it can capture the intrinsic structure embedded in the data. Following the LPP and the spirit of locality constraint, a lot of new dimensionality reduction methods have been proposed recently. In [20], Deng Cai, *et al.* proposed to add the locality constraint in to Nonnegtive Matrix Factorization and propose Locality Preserving Nonnegative Matrix Factorization (LPNMF) to improve the performance of large high dimensional database. Quanquan Gu, *et al.* [21] also focused on the locality preserving property and added it into Weighted Maximum Margin Criterion (WMMC) for text classification, which was called Local Relevance Weighted Maximum Margin Criterion for Text Classification (LRWMMC). In image classification, sparse coding has been proved to be a successful coding method [22]. However, Wang *et al.* [6] argued that the locality preserving constraint was more natural than sparseness constraint and propose Linear Locality Coding (LLC) for coding, which was efficient in coding and achieved the state-of-the-art in image classification.

Inspired by all these works above, we intend to incorporate the locality constraint into KPCA to obtain the optimal dimensionality reduction projection which reduces the reconstruction error and preserves the locality constraint simultaneously. To formalize the locality preserving constraint, a neighborhood graph $\mathbf{W}$ is built and a Laplician Matrix $\mathbf{L}$ is constructed based on the graph.

We add the Laplician Matrix $\mathbf{L}$ into the objective function of KPCA and maximize the new objective function using "*kernel trick*". More details will be illustrated in Sect. 4.

3 A Brief Review of PCA and Kernel PCA

PCA and KPCA is widely used in image understanding and pattern recognition. In this section, a brief review of PCA and KPCA is given in Sects. 3.1 and 3.2, respectively.

3.1 PCA

The aim of PCA is to seek the optimal orthogonal bases of original space to reconstruct the input samples in order to minimize the reconstruction error and compress the data.

Suppose $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}^{D \times N}$ be the centered training set, where D is the dimensionality of the original feature and N is the number of training samples. Let $\mathbf{U} = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_\infty\}$ be the complete orthogonal set, where

$$\mathbf{u}_i^T \mathbf{u}_j = \begin{cases} 1, j = i \\ 0, j \neq i \end{cases} \tag{1}$$

Thus each sample $\mathbf{x}$ can be represented as

$$\mathbf{x} = \sum_{j=1}^{\infty} c_j \mathbf{u}_j \tag{2}$$

If we only adopt a subset of $\mathbf{U}$ to approximate $\mathbf{x}$, which is denoted as $\hat{\mathbf{U}} = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_d\}$, the approximated feature $\hat{\mathbf{x}}$ can be expressed as

$$\hat{\mathbf{x}} = \sum_{j=1}^{d} c_j \mathbf{u}_j \tag{3}$$

As a result, the expected reconstruction error ξ can be represented as

$$\xi = E[(\mathbf{x} - \hat{\mathbf{x}})^T (\mathbf{x} - \hat{\mathbf{x}})] \tag{4}$$

According to (1)–(3), it can be rewritten as

$$\xi = E[\sum_{j=d+1}^{\infty} c_j^2] \tag{5}$$

Because $c_j = \mathbf{u}_j^T \mathbf{x}$, we get

$$\xi = E[\sum_{j=d+1}^{\infty} \mathbf{u}_j^T \mathbf{x} \mathbf{x}^T \mathbf{u}_j] \tag{6}$$

So to minimize ξ, the objective function can be formulated as

$$\hat{\mathbf{U}} = \arg\max_{\hat{\mathbf{U}}}(\frac{1}{2}||\hat{\mathbf{U}}^T\mathbf{X}||_F^2) \quad s.t. \hat{\mathbf{U}}^T\hat{\mathbf{U}} = \mathbf{I} \tag{7}$$

According to the method of Lagrange Multiplier, the optimal $\hat{\mathbf{U}}$ can be obtained by the eigenvalue decomposition of $\mathbf{X}\mathbf{X}^T$.

3.2 Kernel PCA

As is mentioned in section I, traditional PCA only captures the linear embedding relationship of the data, however many data in reality lies in the non-linear embedding space. To dig out the non-linear relationship of the data, Kernel PCA (KPCA) is proposed by Bernhard Scholkopf [15]. It is proved that KPCA performs better than PCA in many problems.

Specifically, suppose Φ be a mapping from original feature space to kernel space which satisfies the Mercer Condition. Thus the inner product of the mapped features $\Phi(x)$, $\Phi(y)$ can be represented as

$$\kappa(\mathbf{x}, \mathbf{y}) = \Phi(\mathbf{x})^T\Phi(\mathbf{y}) \tag{8}$$

Thus the kernel matrix (Gram Matrix) can be represented as

$$\mathbf{K} = [\kappa(\mathbf{x}_i, \mathbf{x}_j)], i, j = 1, 2, \ldots, N \tag{9}$$

And the centered kernel matrix $\hat{\mathbf{K}}$ is

$$\hat{\mathbf{K}} = \mathbf{K} - \mathbf{E}_N\mathbf{K} - \mathbf{K}\mathbf{E}_N + \mathbf{E}_N\mathbf{K}\mathbf{E}_N \tag{10}$$

where $\mathbf{E}_N$ is a $N \times N$ matrix with all elements equals to $\frac{1}{N}$.

Using "*Kernel Trick*" to seek the optimal orthogonal bases in the mapped feature space, which minimizes the reconstruction error, the transformed representation of data $\mathbf{x}$ can be expressed as

$$\mathbf{y} = \mathbf{Q}^T\kappa(\mathbf{X}, \mathbf{x}) \tag{11}$$

where $\mathbf{Q} = [\alpha_1, \alpha_2, \ldots, \alpha_D]$ be the top D eigenvectors of the centered kernel matrix $\hat{\mathbf{K}}$ divided by the square root of the corresponding eigenvalues.

It can be seen that KPCA obtains the linear transformation in a high dimensional kernel space to minimize the reconstruction error using "*Kernel Trick*", which may be a nonlinear transformation in the original space. Thus it can capture the nonlinear relationship in the embedded data space. KPCA is efficient and stable, and is widely used in many areas of signal processing to which the dimensionality reduction is applied.

However, in the derivation of KPCA, it doesn't consider the neighborhood relationship preserving constraint, which is now proven a very important constraint in many related works [6]. It is a very natural and reasonable assumption and we intend to add it into KPCA for better performance.

4 Locality Preserving Kernel PCA

In this section, the mathematical derivation is shown in details.

The objective function of KPCA is

$$\hat{\mathbf{U}} = \arg\max_{\hat{\mathbf{U}}}(\frac{1}{2}||\hat{\mathbf{U}}^T\hat{\Phi}(\mathbf{X})||_F^2) \quad s.t.\ \hat{\mathbf{U}}^T\hat{\mathbf{U}} = \mathbf{I} \tag{12}$$

where $\Phi(\mathbf{X})$ is the zero mean collection of the features in the kernel space. To add the locality constraint into the objective function of KPCA, we first model the neighborhood relationship in the original feature [1] $\mathbf{W}^{N\times N}$, where

$$\mathbf{w}_{ij} = \begin{cases} 1, \text{ if } \mathbf{x}_j \text{ is among the k neighbors of } \mathbf{x}_i \\ 0, \qquad\qquad \text{otherwise} \end{cases} \tag{13}$$

Denoting $\mathbf{y}_i = \mathbf{U}^T\Phi(\mathbf{x}_i)$ as the feature in the transformed space, the locality constraint can be represented as

$$R = \sum_{i,j} ||\mathbf{y}_i - \mathbf{y}_j||_F^2 \mathbf{w}_{ij} \tag{14}$$

Thus the locality constraint can be added into (12) as

$$\hat{\mathbf{U}} = \arg\max_{\hat{\mathbf{U}}}(\frac{1}{2}||\hat{\mathbf{U}}^T\hat{\Phi}(\mathbf{X})||_F^2 - \frac{\lambda}{2}\sum_{i,j} ||\mathbf{y}_i - \mathbf{y}_j||_F^2 \mathbf{w}_{ij}) \quad s.t.\ \hat{\mathbf{U}}^T\hat{\mathbf{U}} = \mathbf{I} \tag{15}$$

where λ is a tradeoff between the reconstruction error and the preservation of locality, bigger λ would increase the credibility of locality. Laplacian Matrix $\mathbf{L}$ is defined as $\mathbf{L} = \mathbf{D} - \mathbf{W}$ [29] with $\mathbf{D}$ is the diagonal matrix $\mathbf{D}_{ii} = \sum_j \mathbf{w}_{ij}$.

Using the Laplacian Matrix $\mathbf{L}$, the objective function of LPKPCA can be rewritten as

$$\hat{\mathbf{U}} = \arg\max_{\hat{\mathbf{U}}}(\frac{1}{2}||\hat{\mathbf{U}}^T\hat{\Phi}(\mathbf{X})||_F^2 - \lambda tr(\mathbf{Y}\mathbf{L}\mathbf{Y}^T)) \quad s.t.\ \hat{\mathbf{U}}^T\hat{\mathbf{U}} = \mathbf{I} \tag{16}$$

where $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N\}$ and $tr()$ is the trace of the matrix.

To obtain the optimal orthogonal bases $\hat{\mathbf{U}}$, the objective function (16) can be rewritten as

$$\hat{\mathbf{U}} = \arg\max_{\hat{\mathbf{U}}}(tr(\hat{\mathbf{U}}^T(\hat{\Phi}(\mathbf{X})\hat{\Phi}(\mathbf{X})^T - \lambda\hat{\Phi}(\mathbf{X})\mathbf{L}\hat{\Phi}(\mathbf{X})^T)\hat{\mathbf{U}})) \tag{17}$$

Thus for each $\mathbf{u}$ in $\hat{\mathbf{U}}$, it satisfies the following objective function

$$\begin{aligned} &\arg\min_{u}(\mathbf{u}^T(\hat{\Phi}(\mathbf{X})\hat{\Phi}(\mathbf{X})^T - \lambda\hat{\Phi}(\mathbf{X})\mathbf{L}\hat{\Phi}(\mathbf{X})^T)\mathbf{u}) \\ &s.t.\ \mathbf{u}^T\mathbf{u} = 1 \end{aligned} \tag{18}$$

According to Lagrange Multiplier method, we get

$$f(\mathbf{u},\mu) = \mathbf{u}^T(\hat{\Phi}(\mathbf{X})\hat{\Phi}(\mathbf{X})^T - \lambda\hat{\Phi}(\mathbf{X})\mathbf{L}\hat{\Phi}(\mathbf{X})^T)\mathbf{u} + \mu(\mathbf{u}^T\mathbf{u} - 1) \tag{19}$$

then

$$\frac{\partial f}{\partial \mathbf{u}} = 2\left(\hat{\Phi}(\mathbf{X})\hat{\Phi}(\mathbf{X})^T - \lambda\hat{\Phi}(\mathbf{X})\mathbf{L}\hat{\Phi}(\mathbf{X})^T\right)\mathbf{u} + 2\mu\mathbf{u} \tag{20}$$

Let $\frac{\partial f}{\partial \mathbf{u}} = 0$, it is easy to see that $\mathbf{u}$ is the eigenvector of $\lambda\hat{\Phi}(\mathbf{X})\mathbf{L}\hat{\Phi}(\mathbf{X})^T - \hat{\Phi}(\mathbf{X})\hat{\Phi}(\mathbf{X})^T$.

Denote $\mathbf{S}^{\Phi} = \hat{\Phi}(\mathbf{X})(\lambda\mathbf{L} - \mathbf{I})\hat{\Phi}(\mathbf{X})^T$, we get

$$\begin{aligned} \mu\mathbf{u} &= \mathbf{S}^{\Phi}\mathbf{u} \\ &= \hat{\Phi}(\mathbf{X})(\lambda\mathbf{L} - \mathbf{I})\hat{\Phi}(\mathbf{X})^T\mathbf{u} \\ &= \hat{\Phi}(\mathbf{X})(\lambda\mathbf{L} - \mathbf{I})\alpha \end{aligned} \tag{21}$$

where we define α as $\hat{\Phi}(X)^T u$.

Thus,

$$\mathbf{u} \propto \hat{\Phi}(\mathbf{X})(\lambda\mathbf{L} - \mathbf{I})\alpha \tag{22}$$

Hence, according to (21) and (22), we get

$$\hat{\mathbf{K}}(\lambda\mathbf{L} - \mathbf{I})\alpha = \mu\alpha \tag{23}$$

Thus α is the eigenvector of $\hat{\mathbf{K}}(\lambda\mathbf{L} - \mathbf{I})$. This α is then divided by a factor ω to satisfy the constraint $\mathbf{u}^T\mathbf{u} = 1$ in Eq. (18). Then the projected feature $\mathbf{y}$ can be represented as

$$\begin{aligned} \mathbf{y} &= \mathbf{U}^T\Phi(\mathbf{x}) \\ &= \mathbf{Q}^T\kappa(\mathbf{X},\mathbf{x}) \end{aligned} \tag{24}$$

where $\mathbf{Q} = [\alpha_1, \alpha_2, \ldots, \alpha_d]_{N\times d}$, and $\kappa(\mathbf{X},\mathbf{x}) = [\kappa(\mathbf{X}_i,\mathbf{x})], i = 1, 2, \ldots, N$. The pseudo code is illustrated in Algorithm 1.

Our LPKPCA can be interpreted as a novel KPCA which captures the intrinsic geometry structure in the data simultaneously. The locality preserving constraint can improve the performance as is illustrated in [6], when the data lies in a low dimensional Riemannian space. Furthermore, our LPKPCA only adds a little bit computation burdens when constructing the neighborhood graph.

Algorithm 1. Locality Preserving Kernel PCA

1: Choose a kernel function κ in (8).
2: Compute $\mathbf{K}$ and $\hat{\mathbf{K}}$ according Eq. (9), (10) using training data.
3: Compute Laplacian matrix $\mathbf{L}$ according to similarity matrix $\mathbf{W}$ in (13).
4: Do the eigenvalue decomposition of $\hat{\mathbf{K}}(\lambda\mathbf{L} - \mathbf{I})$.
5: Sort the eigenvalues in descent order; choose the eigenvectors $\mathbf{v}_i, i = 1, 2, \ldots, d$ corresponding to the top d eigenvalues.
6: Let $\omega_i = \mathbf{v}_i^T(\lambda\mathbf{L} - \mathbf{I})\mathbf{v}_i$, then $\alpha_i = \frac{\mathbf{v}_i}{\sqrt{\omega_i}}$. Thus we get matrix $\mathbf{Q}$ in Eq. (24).

5 Experiments and Results

In this section, we compare the performance of LPKPCA and KPCA on three datasets: ORL dataset [23], Yale Face Database B [24], and Scene 15 dataset [25]. The performance will be judged by average accuracy,

$$Accuracy = \frac{1}{C}\sum_{i=1}^{C} p_i \tag{25}$$

where

$$p_i = \frac{Number\ of\ True\ Positives\ in\ Class\ i}{Total\ Number\ of\ samples\ in\ Class\ i} \tag{26}$$

C is the number of classes. We first describe the experiment settings in Sect. 5.1. And then the results are shown in Sects. 5.2, 5.3 and 5.4, respectively.

5.1 Experiment Preparation

For each dataset, we extract different features according to the content of these datasets. Then the KPCA and LPKPCA are performed on these features. We use cross validation to determine the hyperparameters of k and λ in Eqs. (13) and (15) respectively. When building the similarity matrix $\mathbf{W}$ and Laplacian Matrix $\mathbf{L}$, Euclidean Distance is used in the original space to measure the neighborhood relationship. As for classifiers, Liblinear [26] is adopted, which is a SVM library for large linear classification and can deal with multi-class classification. The parameter of Liblinear is set as follows: $s = 0$, $c = 10$, $e = 0.01$. The RBF Kernel is adopted to compute the kernel matrix in equation (9) with sigma values fit for different datasets. In the experiment, the performance comparison is conducted by evaluation on different number of principle vectors.

5.2 Results on ORL Face Database

There are ten different gray images of each of 40 distinct subjects in the ORL face dataset [23]. These images vary in the different conditions of the lighting, facial expressions (open/closed eyes, smiling/not smiling) and facial details (glasses/no glasses) [23]. Some images in ORL face database are illustrated in Fig. 1. All the

Fig. 1. Some images of ORL face database [23].

images are taken against a dark background and the subjects in an upright, frontal position. The size of each image is 92×112 pixels. Following previous work [27], a gray image is converted to a vector as feature which concatenates all the pixels of the image. We split the dataset with equal number of images as training and testing set, respectively. The comparison performance on ORL dataset with different number of principal vectors can be seen in Fig. 2, where the hyperparameters k and λ are set as 8 and 0.05 respectively by cross-validation. We also show the performance comparison between LPKPCA and KPCA with different λ values in Tables 1 and 2. We can see that the novel LPKPCA outperforms KPCA significantly.

Table 1. Classification accuracy comparison on ORL Face Database with number of principal vector is 60 and $k = 5$

λ	0.02	0.03	0.04	0.05	0.06
LPKPCA %	**94**	**95**	**95**	**96**	**95**
KPCA %	86.5	86.5	86.5	86.5	86.5

5.3 Results on Yale Face Database B

There are 5850 gray face images in Yale Face Database B with 10 subjects. Each subject is seen under 576 viewing conditions (9 poses and 64 illuminations). Moreover, there is one image with ambient illumination (i.e., background) for each pose of the subject. Thus the total number of images for each subject is 585. The images of the 10 individuals are illustrated in Fig. 3 [24]. The size of each

Table 2. Classification accuracy comparison on ORL Face Database with number of principal vector is 60 and $\lambda = 0.05$

k	4	5	6	7	8	9
LPKPCA %	**95**	**96**	**96**	**95.5**	**96.5**	**95**
KPCA %	86.5	86.5	86.5	86.5	86.5	86.5

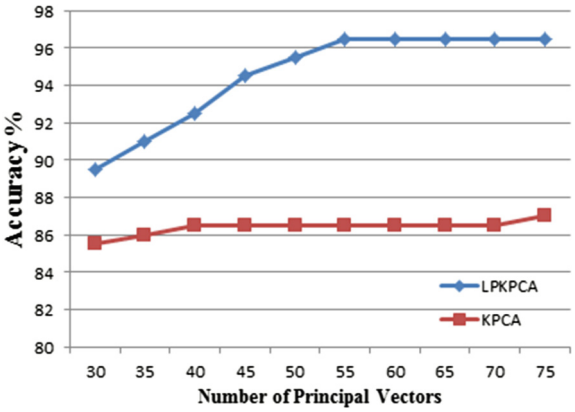

Fig. 2. Classification accuracy on ORL Face Database with different number of principal vectors.

Table 3. Classification accuracy comparison on Yale Face Database B with number of principal vector is 60 and $k = 23$

λ	0.02	0.03	0.04	0.05	0.06
LPKPCA %	**94.83**	**97.79**	**97.90**	**98.38**	**98.66**
KPCA %	92.10	92.10	92.10	92.10	92.10

image is 640×480 and we rescale the images to 40×30. Same as the experiment on ORL, the images are converted to vectors of 1200 dimension by concatenating the pixels. In experiment, the dataset is divided with two parts with equal number which is used as training and testing set, respectively. By cross-validation, the hyperparameters are set as 23 and 0.046 for k and λ respectively. The comparison with different number of principal vectors are shown in Fig. 4 and with different λ values in Tables 3 and 4. We can see that the LPKPCA outperforms KPCA significantly.

5.4 Results on Scene 15 Database

There are 15 kinds of scene image in Scene15 dataset such as store, office, highway, etc. The number of images ranges from 200 to 400 and there are 4485 images

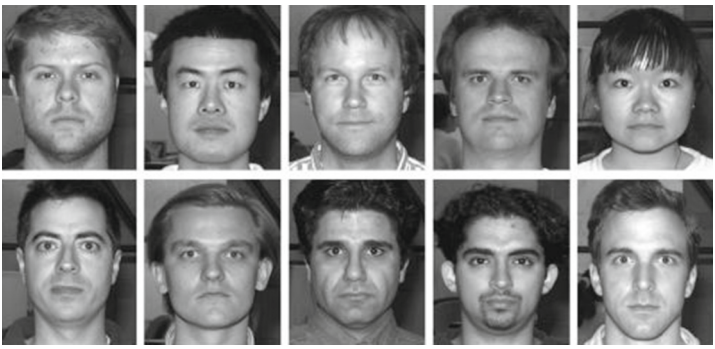

Fig. 3. Some images in Yale Face Database B [24].

Table 4. Classification accuracy comparison on Yale Face Database B with number of principal vector is 60 and $\lambda = 0.046$

k	20	21	22	23	24	25
LPKPCA %	**97.69**	**97.90**	**98.38**	**98.90**	**98.72**	**97.24**
KPCA %	92.10	92.10	92.10	92.10	92.10	92.10

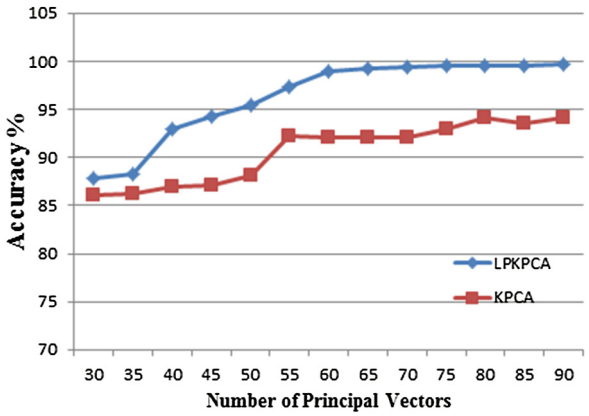

Fig. 4. Classification accuracy on Yale Face Database B with different number of principal vectors.

in total. Some images in Scene 15 database are illustrated in Fig. 5. The dataset is challenging compared to the above ORL and Yale-B dataset because the intra class variance is large. To extract the features to express the holistic content of scene image, GIST descriptor [28] is adopted on all the images. Following the common practice, 100 images are selected randomly per category for training and the remaining ones are treated as test set. The accuracy is reported with different number of principal vectors in Fig. 6. k and λ are set as 8 and 0.038,

Fig. 5. Some illustrations of Scene 15 database [25].

Table 5. Classification accuracy comparison on Scene 15 Database with number of principal vector is 60 and $k = 8$

λ	0.02	0.03	0.04	0.05	0.06
LPKPCA %	**60.35**	**60.27**	**60.81**	**60.49**	**60.03**
KPCA %	59.50	59.50	59.50	59.50	59.50

Table 6. Classification accuracy comparison on Scene 15 Database with number of principal vector is 60 and $\lambda = 0.038$

k	4	5	6	7	8	9
LPKPCA %	**60.29**	**60.23**	**60.41**	**60.57**	**61.05**	**59.86**
KPCA %	59.50	59.50	59.50	59.50	59.50	59.50

respectively, by cross-validation. We also show the performance comparison with different λ in Tables 5 and 6. We can see that the performance of LPKPCA is also better than KPCA.

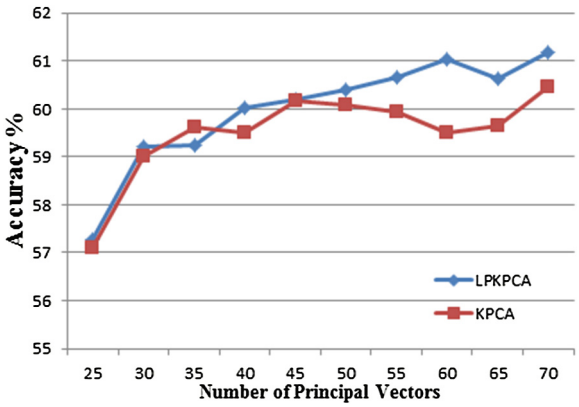

Fig. 6. Classification accuracy on Scene 15 Database with different number of principal vectors.

6 Conclusion

In this paper, a novel kernel PCA approach, which is called Locality Preserving Kernel PCA (LPKPCA), is proposed to simultaneously reduce the reconstruction error in the projected feature space and preserve the neighborhood relationship in the original space. The formulation of LPKPCA is given and experimental results show that LPKPCA achieves better performance on ORL Face Database, Yale Face Database B and Scene 15. The gain in performance results from taking into consideration the intrinsic geometry structure of the data. In the future, we intend to combine the sparseness and locality constraint together to seek for better methods for dimensionality reduction in image understanding and pattern recognition.

References

1. He, X., Niyogi, P.: Locality preserving projections. In: Neural Information Processing Systems, vol. 16, p. 153. MIT (2004)
2. Shen, B., Si, L.: Non-negative matrix factorization clustering on multiple manifolds. In: AAAI (2010)
3. Shen, B., Liu, B.-D., Wang, Q., et al.: SP-SVM: large margin classifier for data on multiple manifolds. In: AAAI Conference on Artificial Intelligence (2015)
4. Liu, B.-D., Wang, Y.-X., Zhang, Y.-J., et al.: Learning dictionary on manifolds for image classification. Pattern Recogn. **46**(7), 1879–1890 (2013)
5. Zheng, Y., Zhang, Y.-J., Larochelle, H.: Topic modeling of multimodal data: an autoregressive approach. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (2014)
6. Wang, J., Yang, J., Yu, K., et al.: Locality-constrained linear coding for image classification. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3360–3367. IEEE (2010)

7. Tenenbaum, J.B., De Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. Science **290**(5500), 2319–2323 (2000)
8. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. Science **290**(5500), 2323–2326 (2000)
9. d'Aspremont, A., El Ghaoui, L., Jordan, M.I., et al.: A direct formulation for sparse PCA using semidefinite programming. SIAM Rev. **49**(3), 434–448 (2007)
10. Ron, Z., Shashua, A.: Nonnegative sparse PCA. In: Schölkopf, B., Platt, J., Hoffman, T. (eds.) Advances in Neural Information Processing Systems. MIT Press, Cambridge (2006)
11. Cands, E.J., Li, X., Ma, Y., et al.: Robust principal component analysis? J. ACM (JACM) **58**(3), 11 (2011)
12. Feng, J., Xu, H., Yan, S.: Online robust PCA via stochastic optimization. In: Advances in Neural Information Processing Systems (2013)
13. Goes, J., Zhang, T., Arora, R., et al.: Robust stochastic principal component analysis. In: Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, pp. 266–274 (2014)
14. Schlkopf, B., Smola, A., Mller, K.-R.: Nonlinear component analysis as a kernel eigenvalue problem. Neural Comput. **10**(5), 1299–1319 (1998)
15. Schlkopf, B., Smola, A., Mller, K.-R.: Kernel principal component analysis. In: Gerstner, W., Hasler, M., Germond, A., Nicoud, J.-D. (eds.) ICANN 1997. LNCS, vol. 1327, pp. 583–588. Springer, Heidelberg (1997)
16. Marukatat, Sanparith: Sparse kernel PCA by kernel K-means and preimage reconstruction algorithms. In: Yang, Qiang, Webb, Geoff (eds.) PRICAI 2006. LNCS (LNAI), vol. 4099, pp. 454–463. Springer, Heidelberg (2006)
17. Tipping, M.E.: Sparse kernel principal component analysis (2001)
18. Honeine, P.: Online kernel principal component analysis: a reduced-order model. IEEE Trans. Pattern Anal. Mach. Intell. **34**(9), 1814–1826 (2012)
19. Belkin, M., Niyogi, P.: Laplacian eigenmaps and spectral techniques for embedding and clustering. In: NIPS, vol. 14 (2001)
20. Cai, D., He, X., Wang, X., et al.: Locality preserving nonnegative matrix factorization. IJCAI **9**, 1010–1015 (2009)
21. Gu, Q., Zhou, J.: Local relevance weighted maximum margin criterion for text classification. In: SDM (2009)
22. Yang, J., Yu, K., Gong, Y., et al.: Linear spatial pyramid matching using sparse coding for image classification. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, pp. 1794–1801. IEEE (2009)
23. Samaria, F.S.: The ORL Database of Faces. http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html
24. Georghiades, A.S., Belhumeur, P.N., Kriegman, D.: From few to many: illumination cone models for face recognition under variable lighting and pose. IEEE Trans. Pattern Anal. Mach. Intell. **23**(6), 643–660 (2001). http://cvc.yale.edu/projects/yalefacesB/yalefacesB.html
25. Li, F.-F., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005, CVPR 2005, vol. 2. IEEE (2005)
26. Fan, R.-E., Chang, K.-W., Hsieh, C.-J., et al.: LIBLINEAR: a library for large linear classification. J. Mach. Learn. Res. **9**, 1871–1874 (2008)
27. Samaria, F.S., Harter, A.C.: Parameterisation of a stochastic model for human face identification. In: Proceedings of the Second IEEE Workshop on Applications of Computer Vision, 1994. IEEE (1994)

28. Aude, O., Torralba, A.: Modeling the shape of the scene: a holistic representation of the spatial envelope. Int. J. Comput. Vis. **42**(3), 145–175 (2001)
29. Gao, S., Tsang, I.W.-H., Chia, L.-T., et al.: Local features are not lonelyLaplacian sparse coding for image classification. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3555–3561. IEEE (2010)