

Research Article

A Kernel Based Neighborhood Discriminant Submanifold Learning for Pattern Classification

Xu Zhao

The Laboratory of Computer Software and Theory, Beijing University of Technology, Beijing 100124, China

Correspondence should be addressed to Xu Zhao; zhaoxul66@gmail.com

Received 18 September 2013; Accepted 15 December 2013; Published 4 February 2014

Academic Editor: Qiankun Song

Copyright © 2014 Xu Zhao. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose a novel method, called Kernel Neighborhood Discriminant Analysis (KNDA), which can be regarded as a supervised kernel extension of Locality Preserving Projection (LPP). KNDA nonlinearly maps the original data into a kernel space in which two graphs are constructed to depict the within-class submanifold and the between-class submanifold. Then a criterion function which minimizes the quotient between the within-class representation and the between-class representation of the submanifolds is designed to separate each submanifold constructed by each class. The real contribution of this paper is that we bring and extend the submanifold based algorithm to a general model and by some derivation a simple result is given by which we can classify a given object to a predefined class effectively. Experiments on the MNIST Handwritten Digits database, the Binary Alphadigits database, the ORL face database, the Extended Yale Face Database B, and a downloaded documents dataset demonstrate the effectiveness and robustness of the proposed method.

1. Introduction

In many practical applications, such as data mining, machine learning, and computer vision, the dimensionality reduction is a necessary preprocessing step for the purpose of noise reduction and reducing the computation complexity. The basic principle of dimensionality reduction is to map the data point from the original space to a low dimensional space through a linear or a nonlinear map. Many dimensionality reduction methods have been developed to deal with this problem. Principal Component Analysis (PCA) [1] and Linear Discriminant Analysis (LDA) [2] are two traditional linear methods. PCA seeks to find a set of bases along which the data exhibit greater variances than other axes. LDA attempts to make the samples as separable as possible in the low dimensional space. The number of features extracted by LDA is at most $c - 1$ (c is the number of classes), which is suboptimal for classification in Bayes sense unless a posteriori probability functions are selected. Since PCA and LDA optimize the mapping based on the global correlations in the given dataset, it is likely to distort the local correlation structures of the data. Meantime, they use the Euclidean distance in the space, which assumes that the space is isotropic

and homogeneous, but this assumption is often invalid due to the curse of dimensionality. So an algorithm which is based on the original Euclidean distance is not a good choice all the time. To address this problem, some other algorithms have been developed, such as Locally Linear Embedding (LLE) [3, 4], Isomap [5], Laplacian Eigenmaps [6], Locality Preserving Projections [7, 8], Neighborhood Preserving Embedding [9], and Tangent Distance Preserving Mapping [10]. These methods attempt to use the local geometry structure of the data manifold to approach the original whole manifold structure. Based on the different geometric intuitions, these methods can reveal the low dimensional structure of the manifold that cannot be detected by classical linear methods.

Recently, lots of novel algorithms which tend to extend or combine the classical linear and nonlinear dimensionality reduction algorithms have been proposed. Weighted Locally Linear Embedding (WLLE) [11] modifies the LLE algorithm based on the weighted distance measurement to improve the dimension reduction and the internal feature extraction performance especially for the deformed distributed data. In reference [12], a novel nonlinear dimensionality reduction algorithm is proposed. It uses relative distance comparisons to explore the local geometrical relations between data points.

All such relative comparisons derived in each neighborhood on the manifold are enumerated and maintained in low-dimensional manifold to be learned. Sparse Multinomial Kernel Discriminant Analysis (sMKDA) [13] is a method for sparse, multinomial kernel discriminant analysis. It is based on the connection between Canonical Variate Analysis (CVA) [14] and least-squares and uses forward selection via orthogonal least-squares to approximate a basis, generalizing a similar approach for binomial problems. Neighborhood Component Analysis (NCA) [15] is a method to learn a transformation by maximizing a stochastic variant of leave-one-out knn score on the training set, such that in the transformed space all similar inputs of the same class are clustered as tightly as possible. There are some other related algorithms, including [16–20], proposed to deal with these kind of problems.

Most of these methods design a heuristic criterion by using the locally geometrical information of the data manifold and through some optimization methods to get the low dimension data. Also, most of these methods may have their supervised versions by using the class label information which is closed to the discriminant ability of an algorithm.

Here, by using a different locally geometrical intuition, we proposed a novel submanifold learning method, called Kernel Neighborhood Discriminant Analysis (KNDA), which is based on the kernel tricks [21–23]. KNDA considers both the within-class submanifold and the betweenclass submanifold by integrating the neighboring information into a weighted matrix in the kernel space. The final goal of KNDA is to keep the within-class data as close as possible in the low dimensional space while keeping away the between-class data in a greatest degree.

However, our method may seemed to be a supervised kernel extension of Locality Preserving Projections (LPP), so before explaining our method, we first briefly introduce LPP.

The rest of the paper is organized as follows. In Section 2, Locality Preserving Projections is reviewed, and then in Section 3, we elaborate the proposed method Kernel Neighborhood Discriminant Analysis (KNDA). Section 4 presents the experiment results. Section 5 will make the discussions about KNDA. Finally, in Section 6, the conclusion is given.

2. Review of Locality Preserving Projections (LPP)

LPP is a linear approximation of the nonlinear Laplacian Eigenmap for learning a locality preserving subspace which preserves the intrinsic geometry of the data and local structure. A neighborhood relationship graph G is built to depict the local structure of the data points.

Let $X = [x_1, x_2, \dots, x_n]$ be a set of vectors, where x_i ($i = 1, 2, \dots, n$) $\in R^m$. LPP seeks a direction P to project X into a low dimensional space in which the local structure of X can be preserved.

The objective function of LPP is defined as

$$\min \sum_i \sum_j \|y_i - y_j\|^2 W_{ij}, \quad (1)$$

where W_{ij} is the weight which depicts the nearness relationship between two data points and y_i and y_j are the mapping results of the data points x_i and x_j .

There are two common ways to compute W_{ij} as follows.

(1) Heat Kernel:

$$W_{ij} = \begin{cases} \exp\left(-\frac{\|x_i - x_j\|^2}{t}\right) & \text{if } x_i(x_j) \text{ is among } k \text{ nearest} \\ & \text{neighbors of } x_j(x_i) \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

Here, parameter $t \in R$. The justification for this choice can be found in [6].

(2) Constant Weight:

$$W_{ij} = \begin{cases} 1 & \text{if } x_i(x_j) \text{ is among } k \text{ nearest} \\ & \text{neighbors of } x_j(x_i) \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

The weight W_{ij} defines a matrix W , whose entry W_{ij} is nonzero only when x_i and x_j are adjacent. Also notice that the entries of W are nonnegative, and W is sparse and symmetric.

There is also an imposed constraint to (1), namely, $Y^T D Y = I$, where D is a diagonal matrix, with $D_{ii} = \sum_j W_{ij}$.

Finally, the minimization problem reduces to the following form:

$$\begin{aligned} \arg \min P^T X L X^T P \\ P^T X D X^T P = I, \end{aligned} \quad (4)$$

where $L = D - W$ is the Laplacian [24] of the graph G constructed by LPP and P is the projection matrix, whose column vectors are the mapping directions by which we can project X into a low dimensional space.

In order to get the optimal projection matrix P_{LPP} , we just need to get the eigenvectors corresponding to the minimum eigenvalues of the generalized eigenvalue problem:

$$X L X^T P = \lambda X D X^T P. \quad (5)$$

When we get P_{LPP} , we can obtain the projection results of X easily by $Y = P_{LPP}^T X$. Here, the size of matrix P_{LPP} is $m * d$, m is the dimensionality of the original space, and d is the dimensionality of the space into which we map the original matrix X .

3. Kernel Neighborhood Discriminant Analysis (KNDA)

Although LPP has a strong principle base and can project the high dimensional data into a low dimensional space with the local geometry structure of the original manifold being preserved, it does not use the class relationship of the data

points, which is more important in pattern classification. Here, we propose a robust and promising method, which first nonlinearly maps the original data points into a high dimensional space and then explicitly constructs an affinity graph by using the label of each point.

Assume that the dataset is represented as $X = [x_1, x_2, \dots, x_n]$, where x_i ($i = 1, 2, \dots, n$) $\in R^m$ and each data point is assigned a label $c_i \in \{1, 2, \dots, N_c\}$, where N_c is the number of classes.

First, we utilize a nonlinear function ϕ to map the original data points into a kernel space H , in which data points can be represented as

$$\phi(X) = [\phi(x_1), \phi(x_2), \dots, \phi(x_n)]. \quad (6)$$

Then, in order to use the label information, we construct the within-class affinity graph and between-class affinity graph in the kernel space H .

3.1. Build the Within-Class Manifold Structure. This step is somewhat like the LPP, we construct an affinity graph, each vertex of which is a data point in the kernel space. But we constrain that there is one edge between two vertexes only when they are from the same class; namely, they have the same label.

When the within-class affinity graph is obtained, each edge of the graph will be given a value to measure the nearness relationship between the two vertexes which form an edge. Here, we use the constant weight:

$$W_{ij} = \begin{cases} 1 & \text{if } \phi(x_i)(\phi(x_j)) \text{ is linked with } \phi(x_j)(\phi(x_i)) \\ 0 & \text{otherwise.} \end{cases} \quad (7)$$

Aiming at preserving the local geometry structure of sub-manifold constructed by each class, we wish to minimize

$$C_{\min} = \sum_i \sum_j (V^T \phi(x_i) - V^T \phi(x_j))^2 W_{ij}, \quad (8)$$

where V is the projection matrix whose columns are the axes of the subspace. Equation (8) can be reduced as follows:

$$\begin{aligned} & \sum_i \sum_j (V^T \phi(x_i) - V^T \phi(x_j))^2 W_{ij} \\ &= 2 \left(\sum_i \sum_j V^T \phi(x_i) W_{ij} \phi(x_i)^T V \right. \\ & \quad \left. - \sum_i \sum_j V^T \phi(x_i) W_{ij} \phi(x_j)^T V \right) \\ &= 2 \left(\sum_i V^T \phi(x_i) D_{ii} \phi(x_i)^T V - V^T \phi(X) W_{\min} \phi(X)^T V \right) \\ &= 2 (V^T \phi(X) D \phi(X)^T V - V^T \phi(X) W_{\min} \phi(X)^T V) \end{aligned}$$

$$\begin{aligned} &= 2 (V^T \phi(X) (D - W_{\min}) \phi(X)^T V) \\ &\xrightarrow{D_{\min}=D} 2 (V^T \phi(X) (D_{\min} - W_{\min}) \phi(X)^T V), \end{aligned} \quad (9)$$

where D is a diagonal matrix with $D_{ii} = \sum_j W_{ij}$ and W_{\min} is a matrix with $W_{\min}(i, j) = W_{ij}$.

Notice that V can be represented as a linear combination of $\phi(x_i)$ ($i = 1, 2, \dots, n$), namely,

$$V = \sum_i \mu_i \phi(x_i) = \phi(X) \mu. \quad (10)$$

So, (8) can be reduced further to

$$\begin{aligned} C_{\min} &= \sum_i \sum_j (V^T \phi(x_i) - V^T \phi(x_j))^2 W_{ij} \\ &= 2 (V^T \phi(X) (D_{\min} - W_{\min}) \phi(X)^T V) \\ &= 2 \mu^T \phi(X)^T \phi(X) (D_{\min} - W_{\min}) \phi(X)^T \phi(X) \mu \\ &= 2 \mu^T K (D_{\min} - W_{\min}) K \mu, \end{aligned} \quad (11)$$

where K is the kernel matrix in the kernel space.

3.2. Build the Between-Class Manifold Structure. The sub-manifolds constructed by each class maybe overlap, which is the main reason why the recognition rate is low in the classification works. So, how to separate the submanifolds in the low dimensional space is a key issue in dimensionality reduction researches. Here, we explore the between-class relationship to build an affinity graph which reflects the membership between data points with different labels.

We link two points $\phi(x_i)$ and $\phi(x_j)$ in the kernel space only when the Euclidean distance between them is less than some limit values and they are from different class. Here, unlike building the within-class manifold structure, we add another limit; namely, the two data points with different labels must reside on a local sphere with a small radius. We suppose this limit is because if two data points with different labels are much close to each other in the space, there will be a higher possibility of classifying the data points falsely. So we are compelled to add this limit in order to make the points with different labels far away furthest in the low dimension space. The radius value should be chosen to make the local spheres contains a few data points that have different labels to the data point reside on the centre of the sphere, meanwhile, the radius value should be as small as possible, which will reflect the local structure of the manifold faithfully.

When the between-class affinity graph is obtained, like in Section 3.1, we give each edge of the graph a value. We also utilize the constant weight here; namely,

$$W_{ij} = \begin{cases} 1 & \text{if } \|\phi(x_i) - \phi(x_j)\| \leq \varepsilon \\ & \text{and label } (\phi(x_i)) \neq \text{label } (\phi(x_j)) \\ 0 & \text{otherwise.} \end{cases} \quad (12)$$

Since we conduct this step in the kernel space, so we can calculate $\|\phi(x_i) - \phi(x_j)\|$ like the following:

$$\begin{aligned} & \|\phi(x_i) - \phi(x_j)\| \\ &= \sqrt{\phi(x_i) - \phi(x_j)^T (\phi(x_i) - \phi(x_j))} \\ &= \sqrt{(\phi(x_i), \phi(x_i)) - 2(\phi(x_i), \phi(x_j)) + (\phi(x_j), \phi(x_j))} \\ &= \sqrt{K(i, i) - 2K(i, j) + K(j, j)}, \end{aligned} \quad (13)$$

where matrix K is the kernel matrix whose entries are the inner product of all pairs of $\phi(x_i)$ and $\phi(x_j)$ ($i, j = 1, 2, \dots, n$).

In order to make the between-class submanifolds constructed by each class as separable as possible in the low dimensional space, we wish to maximize

$$C_{\max} = \sum_i \sum_j (V^T \phi(x_i) - V^T \phi(x_j))^2 W_{ij}. \quad (14)$$

According to Section 3.1, (14) can be reduced to

$$\begin{aligned} C_{\max} &= \sum_i \sum_j (V^T \phi(x_i) - V^T \phi(x_j))^2 W_{ij} \\ &= 2\mu^T K (D_{\max} - W_{\max}) K \mu, \end{aligned} \quad (15)$$

where D_{\max} is a diagonal matrix with $D_{ii} = \sum_j W_{ij}$, W_{\max} is a matrix with $W_{\max}(i, j) = W_{ij}$, and K is the kernel matrix in the kernel space.

3.3. Criterion Function. Since Kernel Neighborhood Discriminant Analysis is designed to preserve the within-class geometry structure of all the submanifolds and meanwhile keep away the between-class submanifold, we define the criterion function as follows:

$$\begin{aligned} \arg \min &= \frac{C_{\min}(V)}{C_{\max}(V)} \\ &= \frac{\mu^T K (D_{\min} - W_{\min}) K \mu}{\mu^T K (D_{\max} - W_{\max}) K \mu}. \end{aligned} \quad (16)$$

This problem can be reformulated as a constrained minimization problem:

$$\begin{aligned} \min & \mu^T K (D_{\min} - W_{\min}) K \mu \\ \text{s.t.} & \mu^T K (D_{\max} - W_{\max}) K \mu = 1. \end{aligned} \quad (17)$$

We can obtain the solution of (17) easily by solving the following generalized eigenvalue problem:

$$K (D_{\min} - W_{\min}) K \mu = \lambda K (D_{\max} - W_{\max}) K \mu. \quad (18)$$

If $K(D_{\max} - W_{\max})K$ is invertible, we can reduce (18) to the common eigenvalue problem:

$$(K (D_{\max} - W_{\max}) K)^{-1} K (D_{\min} - W_{\min}) K \mu = \lambda \mu. \quad (19)$$

In real operation, we can add a diagonal matrix with small entry value, such as 0.01, to the matrix $K(D_{\max} - W_{\max})K$ to ensure that it has full rank. Then the optimal solution vectors μ_* of (17) are the eigenvectors corresponding to the smallest d eigenvalues of $(K(D_{\max} - W_{\max})K)^{-1} K(D_{\min} - W_{\min})K$ (d is the reduction dimensionality).

When μ_* is gotten, the data points in the kernel space can be mapped into a subspace by

$$Y = \mu_*^T K. \quad (20)$$

And for a new test sample x , it can be mapped to the subspace by

$$y = \mu_*^T \phi(X)^T \phi(x). \quad (21)$$

Algorithm 1. The formal algorithm procedure can be described as follows.

Let $X = [x_1, x_2, \dots, x_n]$ and x_i ($i = 1, 2, \dots, n$) $\in R^m$ be the dataset.

Step 1. Compute the kernel matrix K of the data points.

Step 2. Construct the within-class affinity graph, and compute the weigh matrix W_{\min} and D_{\min} :

$$w_{ij} = \begin{cases} 1 & \text{if } \phi(x_i) \text{ is linked with } \phi(x_j) \\ 0 & \text{otherwise,} \end{cases} \quad (22)$$

D_{\min} is a diagonal matrix with $D_{ii} = \sum_j W_{ij}$.

Step 3. Construct the between-class affinity graph, and compute the weigh matrix W_{\max} and D_{\max} :

$$w_{ij} = \begin{cases} 1 & \text{if } \|\phi(x_i) - \phi(x_j)\| \leq \varepsilon \\ & \text{and label } (\phi(x_i)) \neq \text{label } (\phi(x_j)) \\ 0 & \text{otherwise,} \end{cases} \quad (23)$$

D_{\max} is a diagonal matrix with $D_{ii} = \sum_j W_{ij}$.

Step 4. Solve the eigenvalue problem:

$$(K (D_{\max} - W_{\max}) K)^{-1} K (D_{\min} - W_{\min}) K \mu = \lambda \mu. \quad (24)$$

In real operation, we can add a diagonal matrix with small entry value, such as 0.01, to the matrix $K(D_{\max} - W_{\max})K$ to ensure that it has full rank.

Step 5. Produce the mapped vectors. Let $\mu_1^*, \mu_2^*, \dots, \mu_d^*$ be the eigenvectors corresponding to the smallest d eigenvalues of the eigenvalue problem shown in Step 4; the final d dimension embedding result of the original data points is

$$Y = \mu_*^T K, \quad (25)$$

where $\mu_* = [\mu_1^*, \mu_2^*, \dots, \mu_d^*]$.

4. Experiments and Discussions

In this section, we conduct several experiments on different datasets to demonstrate the effectiveness and the robustness



FIGURE 1: Sample digit images from MNIST database.

of our proposed method Kernel Neighborhood Discriminant Analysis (KNDA). Remember that in all the following experiments the parameter k used to find the nearest neighborhoods in LPP and KLPP is set to 10. Also by an auxiliary experiment, we find that by using polynomial kernel with KNDA can get better results than other kernel functions, such as Gaussian kernel and sigmoid kernel, and there are no serious impacts on the experiment results when parameter d is disturbed in polynomial kernel. So for comparison and simplicity, we employ polynomial kernel by choosing “ d ” equal to 2 in KNDA, KPCA, KLDA, and KLPP.

4.1. Digit Visualization

4.1.1. MNIST Handwritten Digits. In this experiment, we use a subset of the MNIST Handwritten Digits database [25], which contains 8 bit grayscale images of “0” through “9”. Each digit image sample is represented as a high dimensional vector of length 784. Figure 1 depicts two digits “0” and “1” of the database.

We choose randomly 100 samples of every digit to form our dataset, and we compare KNDA with SLPP, SKLPP, PCA, and KPCA. The results are shown in Figure 2.

We can see that the embedding result of KNDA almost separates each class of the sample perfectly, while all the other four methods result in the overlap of the data points from different classes. That is because PCA type methods are designed to preserve the global structure of the manifold and LPP type methods consider the local structure of the manifold more. Also PCA and KPCA neglect the key factor “class label” that is very important in the classification work. Notice that digits “4” and “9” are not separated completely by KNDA (see Figure 2). If we look back to the digits “4” and “9” in the original database, we can find that the two digits in the database are somewhat similar. The images are shown in Figure 3.

Here, we look back to Figure 2. We can find that the centroid of the two clusters of digits “4” and “9” is too close, if we can make the abscissa of the two centroid much separated, maybe we can split digits “4” and “9” totally. So, we discard the eigenvector corresponding to the first smallest eigenvalue of matrix $(K(D_{\max} - W_{\max})K)^{-1}K(D_{\min} - W_{\min})K$ and use the two eigenvectors corresponding to the second and the third smallest eigenvalues of matrix $(K(D_{\max} - W_{\max})K)^{-1}K(D_{\min} - W_{\min})K$. This means that we use the original x -direction as the y -direction now. The experiment result is in Figure 4.

This time, digits “4” and “9” are separated completely. The result can be seen as a 90° globally clockwise rotation to the result contained by KNDA in Figure 3 at a certain degree, and now notice that the y -direction distance of the cluster centroids of digits “4” and “9” is almost the same as

the x -direction in Figure 3, which is because now the eigenvector corresponding to the 2nd eigenvalue of matrix $(K(D_{\max} - W_{\max})K)^{-1}K(D_{\min} - W_{\min})K$ is used as the embedding vector of y -direction, and it is used as the embedding vector of x -direction in the foregoing experiment.

4.1.2. Binary Alphadigits. The dataset used in this part is Binary Alphadigits [25], which is composed of Binary 20×16 digits of “0” to “9” and capital “A” to “Z”. Each digit has 39 examples. Here, we use all the samples of the database. Figure 5 depicts digit “9” and character “A” of the database.

By KNDA, we map all the data points to the 2-dimension space, and the experiment results are shown in Figure 6.

It is very interesting and exciting that the samples from each class are embedded to a point in the 2-dimension space, which demonstrates the effectiveness and robustness of KNDA. Since this database contains 36 different classes, namely, digits “1-9” and characters “A-Z”, we do not show the results produced by PCA, LPP, and NPE. All these three methods’ results are somewhat like the ones in the MNIST database.

However, digit “0” and character “O” are overlapped, which is not deviant, if we take a look at the images in the database, which is shown in Figure 7.

They are the same! That is why KNDA embeds digit “0” and character “O” in the same place in the 2-dimension embedding space.

4.2. Face Recognition

4.2.1. ORL. In this experiment, we use a famous face database ORL [26]. ORL contains 40 different subjects, and each subject has ten different images. The images include variation in facial expressions (smile or not and open/closed eyes) and pose. Figure 8 illustrates two sample subjects of the ORL database along with variations in facial expressions and pose.

We test KNDA against DNDA (Direct Neighborhood Discriminant Analysis), PCA, KPCA, LPP, KLPP, LDA, and KLDA to demonstrate the predominance of KNDA. DNDA is conducted by directly building the within-class manifold structure and the within-class manifold structure in the original data space and the other steps are the same as KNDA, so we do not explain DNDA at length here. We form the training set by randomly selecting 5 images per individual with labels, and the rest of the database was considered to be the testing set. Nearest neighborhood classifier is employed in the experiments. Also, we find that if we run PCA prior to LPP, the results of LPP can increase a lot. So in our experiment, before conducting LPP we first run PCA. The experiments are conducted 10 times, and we report the average results here. The results are given in Figure 9. The horizontal axis represents the dimension of the subspace and the vertical axis stands for the recognition rate.

It can be seen that KNDA outperforms all the other 7 methods. Although DNDA, PCA, KPCA, LDA, KLDA, and LPP perform well, KNDA exceeds them a lot. Another point we should pay attention to is that LPP has the same performance when the dimension is larger than 50. This is

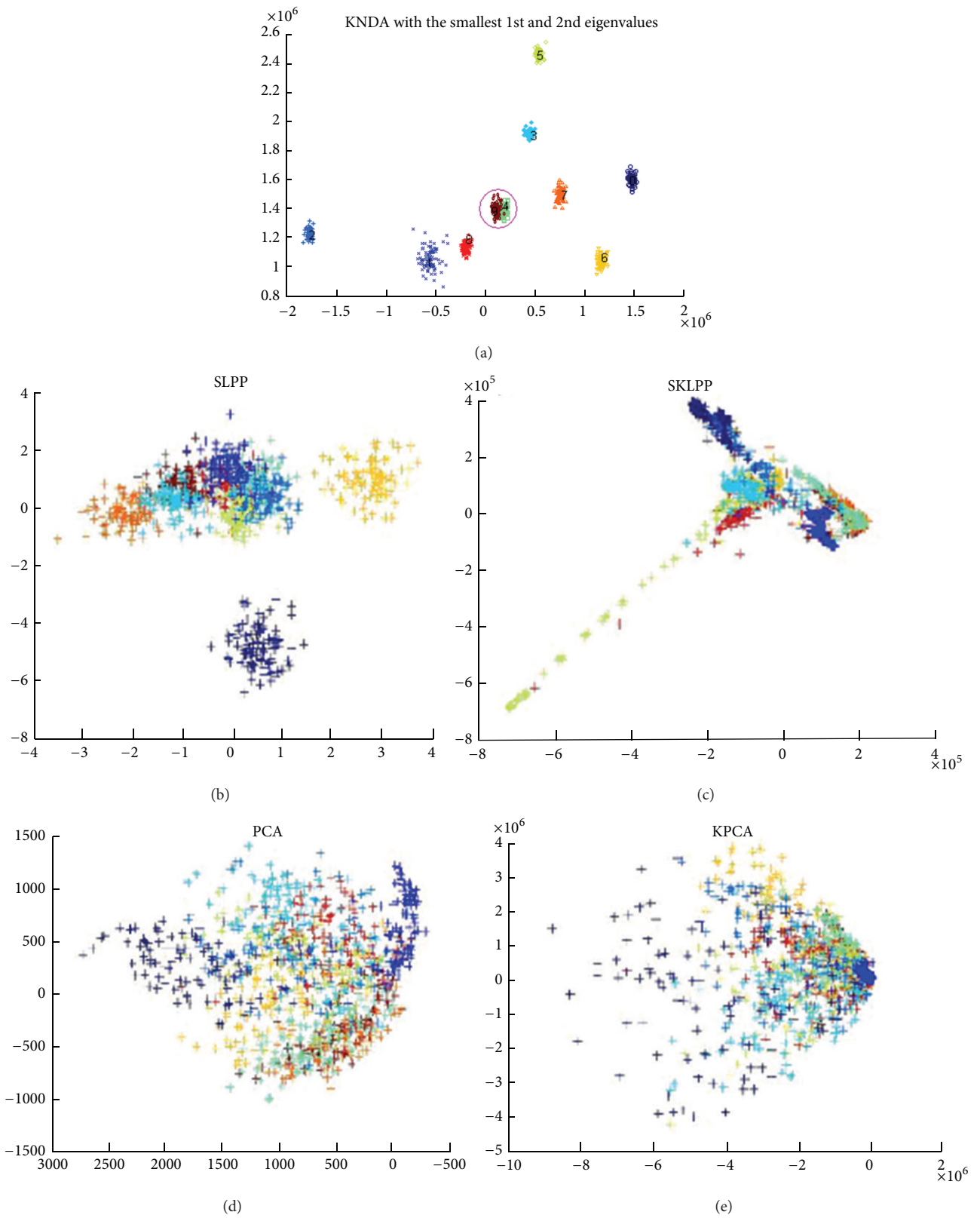


FIGURE 2: The 2-dimension embedding results by KNDA, SLPP, SKLPP, PCA, and KPCA using MNIST database.

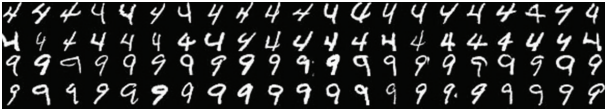


FIGURE 3: Subset of digits “4” and “9” in the MNIST database.



FIGURE 5: Subset of digit “9” and character “A” in the Binary Alphadigits database.

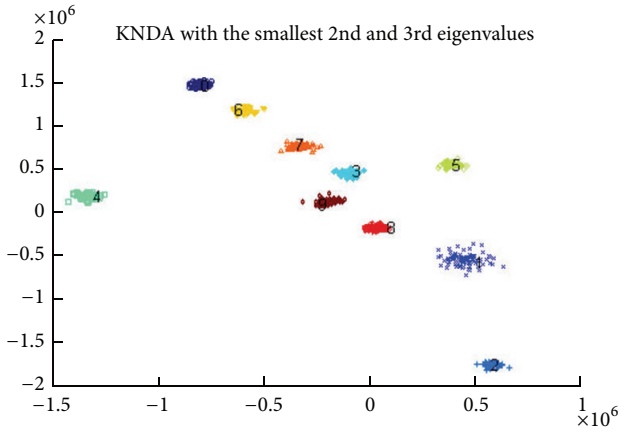


FIGURE 4: The 2-dimension embedding result by KNDA using eigenvectors corresponding to the 2nd and the 3rd smallest eigenvalues of matrix $(K(D_{\max} - W_{\max})K)^{-1}K(D_{\min} - W_{\min})K$.

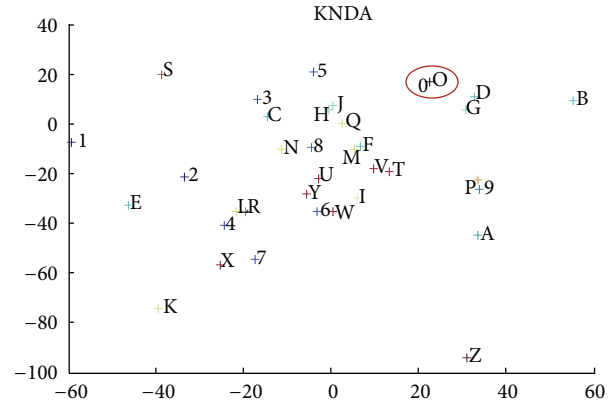


FIGURE 6: The 2-dimension embedding results by KNDA, using Binary Alphadigits database.

TABLE 1: The best recognition rate achieved by each method along with the corresponding subspace dimension d .

	Subspace dimension d	Recognition rate (%)
DNDA	120	93.8
KNDA	90	95.55
PCA	70	85.35
KPCA	90	87.8
LPP	40	86.6
KLPP	120	79.55
LDA	39	90.2
KLDA	39	92.1



FIGURE 7: Subset of digit “0” and character “O” in the Binary Alphadigits database.



FIGURE 8: Sample face images from the ORL database along with variations in facial expressions and poses.

because we run PCA to preserve 90 percent principal component prior to conducting LPP, and the subspace dimension obtained by PCA is smaller than 50. Also except KDNA, DNDA, LDA, and KLDA, all the other 4 methods do not consider explicitly the class labels of the data points, so when comparing with KNDA, they look poorer. Meantime, although it is obvious that DNDA, LDA, and KLDA, which consider the label information explicitly, get better results, KNDA outperforms them at a certain degree. We also list the best recognition rate achieved by each method along with the corresponding subspace dimension d in Table 1.

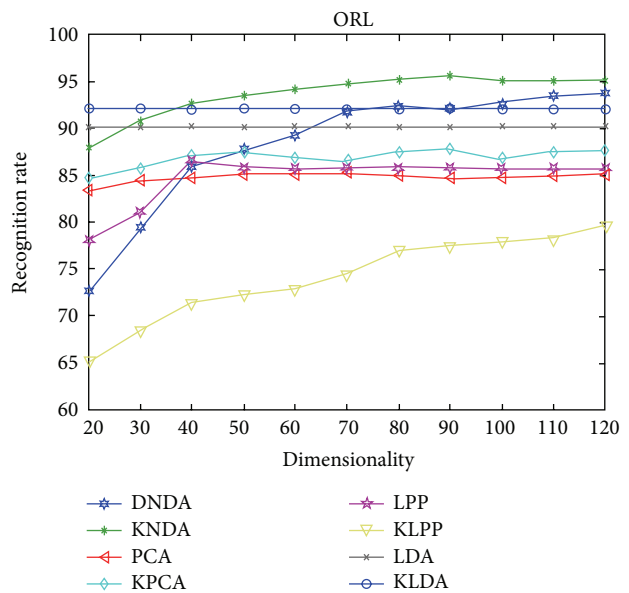


FIGURE 9: The recognition rates versus the subspace dimension on ORL database.

4.2.2. *Extended Yale Face Database B.* In this section, Extended Yale Face Database B [27] is used to conduct our experiments. Extended Yale Face Database B contains 16128 images of 28 human subjects under 9 poses and 64 illumination conditions. We use a subset of Yale Database B here which contains 38 subjects under different illumination



FIGURE 10: Cropped example face images from the Extended Yale Face Database B database under different illuminations.

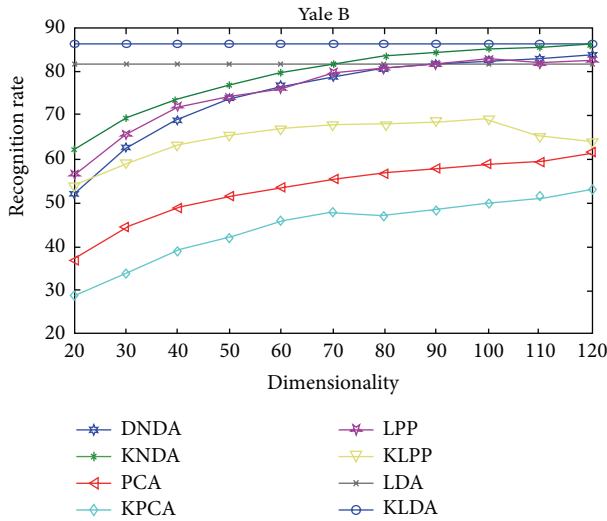


FIGURE 11: The recognition rates versus the subpace dimension on Extended Yale Face Database B.

conditions and each subject has 64 cropped images. Figure 10 illustrates a cropped example face images from the Extended Yale Face Database B, database under different illuminations.

We also test KNDA against DNDA (Direct Neighborhood Discriminant Analysis), PCA, KPCA, LPP, KLPP, LDA, and KLDA. The training set is formed by randomly selecting 30 images per subject with labels, and the rest of the database was considered to be the testing set. Nearest neighborhood classifier is employed in the experiments. Like in Section 4.2.1, we run PCA prior to LPP to achieve the best performance of LPP. The experiments are conducted 10 times, and we report the average results here. The results are given in Figure 11. The horizontal axis represents the dimension of the subpace and the vertical axis stands for the recognition rate.

This time, KNDA outperforms almost all the other 7 methods too. Although KLDA gets a better result when the dimensionality is low, KNDA exceeds KLDA for 0.11 percent when the dimension equals 120. Also, we must mention that this result may be related to the dataset. In Section 4.2.1 we can see that KNDA outperforms KLDA a lot and the intrinsic relationship between the dataset and the algorithm is still unknown. Meanwhile, we can see that LPP is better than PCA, KPCA, and KLPP, but KNDA exceeds it nearly by 3 percent during the increasing of the dimension. Also, if we look back to the experiment results in Section 4.2.1, we will find that this time KLPP exceeds PCA and KPCA a lot, but in the ORL database, this is almost the reverse. However, KNDA gets the best recognition rate in both

TABLE 2: The best recognition rate achieved by each method along with the corresponding subpace dimension d .

Method	Subspace dimension d	Recognition rate (%)
DNDA	120	83.5
KNDA	120	86.36
PCA	120	61.22
KPCA	120	52.82
LPP	100	82.78
KLPP	100	68.92
LDA	27	81.5
KLDA	27	86.25

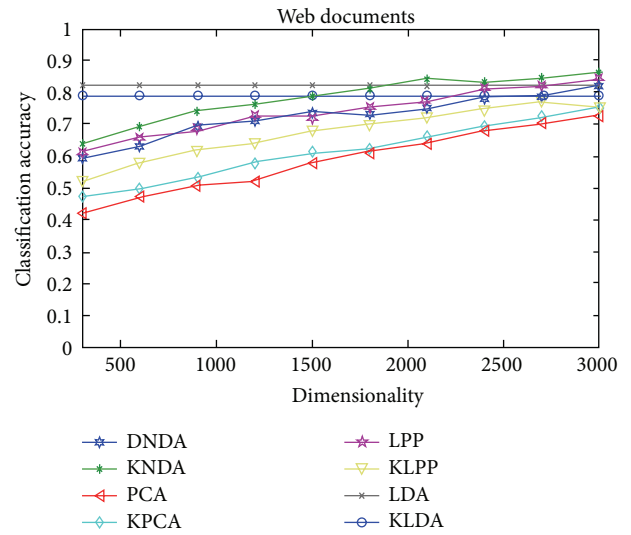


FIGURE 12: The recognition rates versus the subpace dimension on random downloaded Web documents which belonged to 25 different classes.

the ORL database and the Extended Yale Face Database B, which indicates a stabilized performance.

The best recognition rate achieved by each method along with the corresponding subpace dimension d is shown in Table 2.

4.3. Text Categorization. In order to test the performance of KNDA in the real application, a random set of Web documents that belonged to 25 different classes has been downloaded from the Internet by an auxiliary Web crawler program. The number of the documents belonging to different classes is very different. Some classes, such as sport or computer, have more than 2000 documents, and on the other hand, classes such as decoration have less than 500 documents. This difference is deliberately imported to simulate the real application context to the maximum extent. We use 80 percent of each class to compose the train set and the other 20 percent of each class are used to test. The experiments are conducted 10 times, and we report the average results here. The results are given in Figure 12. The horizontal axis represents the dimension of the subpace and the vertical axis stands for the recognition rate.

TABLE 3: The best recognition rate achieved by each method along with the corresponding subspace dimension d .

	Subspace dimension d	Recognition rate (%)
DNDA	3000	82.13
KNDA	3000	86.24
PCA	3000	72.53
KPCA	3000	75.01
LPP	3000	84.2
KLPP	2700	77.11
LDA	24	82.01
KLDA	24	78.75

We can see clearly that KNDA is the best method globally comparing with others, although it is not the best one when the dimensionality is low. Also, this time LPP gets a better performance than lots of the other methods, which suggests that LPP is not stable when using different dataset. On the other hand, since DNDA and KNDA consider both the within-class structure and the between-class structure of the data manifold, a more stable performance is gotten. Meantime, remember that we use randomly downloaded Web documents dataset as the train set, and the distribution of the documents is not uniform, which means that KNDA can get a more robust and effective result in the real application context.

Also, the best recognition rate achieved by each method along with the corresponding subspace dimension d is shown in Table 3.

5. Discussions

KNDA is a nonlinear dimension reduction algorithm which is based on the graph theory. There are other graph based algorithms, such as S-Isomap [28], Tensor Subspace Analysis [29], and Conformal Embedding Analysis (CEA) [30].

The key difference between KNDA and them is how to construct the graph and how to choose the optimization criterion, which is very important to the algorithm. KNDA uses not only the within-class information but the between-class information to construct the graph which can effectively improve the discriminant ability of an algorithm.

Meantime, the criterion function of KNDA is similar to the LDA type algorithms, so here we briefly summarize the similarity and dissimilarity between them. The LDA type algorithms are based on the global correlations in the given dataset; it is likely to distort the local correlation structures of the data. On the contrary, KNDA fully uses the locally geometrical information of the data manifold to construct the discriminant criterion and can adapt to the problem of the curse of dimensionality.

Also, one can combine both the global and the local information of the data to design an algorithm, such as [31], in which a new algorithm called Distinguishing Variance Embedding (DVE) is proposed. DVE unfolds the dataset by maximizing the global variance subject to the proximity relation preservation constraint originated in Laplacian Eigenmaps.

6. Conclusions

In this paper, a kernel based neighborhood discriminant submanifold learning algorithm called Kernel Neighborhood Discriminant Analysis (KNDA) is proposed. KNDA is derived by first nonlinearly mapping the original dataset into a kernel space, and then within-class submanifold and between-class manifold are modeled in the kernel space in order to separate each submanifold constructed by each class. Through solving an eigenvalue problem, we get the embedding vectors in the low dimension space. Digit visualization, face recognition, and a real documents' categorization experiments are conducted on several different artificial or real dataset to demonstrate the dominance of KNDA.

Conflict of Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

References

- [1] I. T. Jolliffe, *Principal Component Analysis*, Springer, New York, NY, USA, 1986.
- [2] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces versus fisherfaces: recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.
- [3] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
- [4] L. K. Saul and S. T. Roweis, "Think globally, fit locally: unsupervised learning of low dimensional manifolds," *Journal of Machine Learning Research*, vol. 4, no. 2, pp. 119–155, 2003.
- [5] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.
- [6] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Proceedings of the Advances in Neural Information Processing Systems 14*, Vancouver, Canada, 2002.
- [7] X. He and P. Niyogi, "Locality preserving projections," in *Proceedings of the Advances in Neural Information Processing Systems*, Vancouver, Canada, 2003.
- [8] X. He, S. Yan, Y. Hu, P. Niyogi, and H.-J. Zhang, "Face recognition using Laplacianfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 328–340, 2005.
- [9] X. He, D. Cai, S. Yan, and H.-J. Zhang, "Neighborhood preserving embedding," in *Proceedings of the 10th IEEE International Conference on Computer Vision (ICCV '05)*, vol. 2, pp. 1208–1213, Beijing, China, October 2005.
- [10] X. Zhao and Z. Jiang, "A nonlinear structure preserving dimensionality reduction algorithm," in *Proceedings of the 10th IASTED International Conference on Signal Processing and Applications*, pp. 200–204, 2008.
- [11] Y. Pan, S. S. Ge, and A. Al Mamun, "Weighted locally linear embedding for dimension reduction," *Pattern Recognition*, vol. 42, no. 5, pp. 798–811, 2009.
- [12] C. X. Zhang, S. M. Xiang, F. P. Nie, and Y. Q. Song, "Nonlinear dimensionality reduction with relative distance comparison," *Neurocomputing*, vol. 72, no. 7–9, pp. 1719–1731, 2009.

- [13] R. F. Harrison and K. Pasupa, "Sparse multinomial kernel discriminant analysis (sMKDA)," *Pattern Recognition*, vol. 42, no. 9, pp. 1795–1802, 2009.
- [14] N. A. Campbell, "Canonical variate analysis—a general model formulation," *The Australian Journal of Statistics*, vol. 26, no. 1, pp. 86–96, 1984.
- [15] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov, "Neighborhood components analysis," in *Proceedings of the Advances in Neural Information Processing Systems*, Cambridge, Mass, USA, 2004.
- [16] F. Nie, S. Xiang, and C. Zhang, "Neighborhood minmax projections," in *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI '07)*, pp. 993–998, Hyderabad, India, January 2007.
- [17] C. Zhang, F. Nie, and S. Xiang, "A general kernelization framework for learning algorithms based on kernel PCA," *Neurocomputing*, vol. 73, no. 4–6, pp. 959–967, 2010.
- [18] F. Nie, S. Xiang, Y. Song, and C. Zhang, "Orthogonal locality minimizing globality maximizing projections for feature extraction," *Optical Engineering*, vol. 48, no. 1, Article ID 017202, 5 pages, 2009.
- [19] F. Nie, S. Xiang, Y. Song, and C. Zhang, "Extracting the optimal dimensionality for local tensor discriminant analysis," *Pattern Recognition*, vol. 42, no. 1, pp. 105–114, 2009.
- [20] Y. Song, F. Nie, and C. Zhang, "Semi-supervised sub-manifold discriminant analysis," *Pattern Recognition Letters*, vol. 29, no. 13, pp. 1806–1813, 2008.
- [21] V. N. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, New York, NY, USA, 1998.
- [22] B. Schölkopf and A. J. Smola, *Learning with Kernels*, The MIT Press, Cambridge, Mass, USA, 2002.
- [23] J. Ham, D. D. Lee, S. Mika, and B. Schölkopf, "A kernel view of the dimensionality reduction of manifolds," in *Proceedings of the 21st International Conference on Machine Learning (ICML '04)*, p. 47, Banff, Canada, 2004.
- [24] F. R. K. Chung, *Spectral Graph Theory*, vol. 92 of *CBMS Regional Conference Series in Mathematics*, 1997.
- [25] <http://www.cs.toronto.edu/~roweis/data.html>.
- [26] F. S. Samaria and A. C. Harter, "Parameterisation of a stochastic model for human face identification," in *Proceedings of the 2nd IEEE Workshop on Applications of Computer Vision*, pp. 138–142, December 1994.
- [27] A. S. Georghiadis, P. N. Belhumeur, and D. J. Kriegman, "From few to many: illumination cone models for face recognition under variable lighting and pose," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 643–660, 2001.
- [28] X. Geng, D.-C. Zhan, and Z.-H. Zhou, "Supervised nonlinear dimensionality reduction for visualization and classification," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 35, no. 6, pp. 1098–1107, 2005.
- [29] X. He, D. Cai, and P. Niyogi, "Tensor subspace analysis," in *Proceedings of the Advances in Neural Information Processing Systems*, Vancouver, Canada, December 2005.
- [30] Y. Fu, M. Liu, and T. S. Huang, "Conformal embedding analysis with local graph modeling on the unit hypersphere," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop on Component Analysis in Computer Vision (CVPR '07)*, pp. 1–6, Minneapolis, Minn, USA, June 2007.
- [31] Q. G. Wang and J. W. Li, "Combining local and global information for nonlinear dimensionality reduction," *Neurocomputing*, vol. 72, no. 10–12, pp. 2235–2241, 2009.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

