Pattern Recognition 44 (2011) 1387–1402

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

From classifiers to discriminators: A nearest neighbor rule induced discriminant analysis

Jian Yang [a,*], Lei Zhang [b], Jing-yu Yang [a], David Zhang [b]

[a] School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing 210094, PR China
[b] Department of Computing, Hong Kong Polytechnic University, Kowloon, Hong Kong

ARTICLE INFO

Article history:
Received 15 February 2009
Received in revised form
10 November 2010
Accepted 14 January 2011
Available online 21 January 2011

Keywords:
Classification
Classifier
Feature extraction
Discriminant analysis
Dimensionality reduction
Pattern recognition

ABSTRACT

The current discriminant analysis method design is generally independent of classifiers, thus the connection between discriminant analysis methods and classifiers is loose. This paper provides a way to design discriminant analysis methods that are bound with classifiers. We begin with a local mean based nearest neighbor (LM-NN) classifier and use its decision rule to supervise the design of a discriminator. Therefore, the derived discriminator, called local mean based nearest neighbor discriminant analysis (LM-NNDA), matches the LM-NN classifier optimally in theory. In contrast to that LM-NNDA is a NN classifier induced discriminant analysis method, we further show that the classical Fisher linear discriminant analysis (FLDA) is a minimum distance classifier (i.e. nearest Class-mean classifier) induced discriminant analysis method. The proposed LM-NNDA method is evaluated using the CENPARMI handwritten numeral database, the NUST603 handwritten Chinese character database, the ETH80 object category database and the FERET face image database. The experimental results demonstrate the performance advantage of LM-NNDA over other feature extraction methods with respect to the LM-NN (or NN) classifier.

© 2011 Elsevier Ltd. All rights reserved.

1. Introduction

The nearest neighbor (1-NN) classifier is one of the most widely used classifiers due to its simplicity and effectiveness. Cover and Hart laid the theoretical foundation of 1-NN classifier and showed that when the training sample size approaches to infinity, the error rate of the NN classifier is bounded above by twice the Bayes error rate in 1967 [1]. As a generalization of 1-NN classifier, *K*-NN classifier was presented subsequently [2]. In recent years, with the popularity of manifold learning, the NN-based classification methods arouse considerable research interests and a number of improved variants of the NN classifier have been developed [3–8]. Among the most simple and interesting is the local mean based nearest neighbor (LM-NN) classifier, which uses the mean of the *R* nearest neighbors within a class as the prototype of the class [8]. The LM-NN classifier was demonstrated to be more robust to outliers than the classical 1-NN and *K*-NN classifiers and thus achieves better classification performance [8].

If the dimension of the observation space is very high, it is generally time-consuming to perform the NN-based classification directly based on all of the original features. In addition, using all features for classification is not necessary to achieve optimum results due to the problem of "dimensionality curse". Therefore, we usually perform feature extraction (or dimensionality reduction) first before the classification step. Discriminant analysis is a fundamental tool for feature extraction. By far, numerous discriminant analysis methods have been developed. Among the most well-known is Fisher linear discriminant analysis (FLDA) [36]. FLDA seeks to find a projection axis such that the Fisher criterion (i.e. the ratio of the between-class scatter to the within-class scatter) is maximized after the projection of samples. FLDA receives intense attention in the past decade and numerous FLDA variants were put forward to deal with real-world small sample size problems [9–16]. A nonlinear version of FLDA, the kernel Fisher Discriminant (KFD), was proposed for dealing with the data with nonlinear structures [17–20]. Another nonlinear discriminant analysis version, Locally Linear Discriminant Analysis which involves a set of locally linear transformations, was presented recently based on the idea that global nonlinear data structures are locally linear [21]. In addition, motivated by the idea of manifold learning algorithms [37,38], researchers designed a family of locality characterization based discriminant analysis techniques, such as Locality Preserving Projections (LPP) [22], Local discriminant embedding [23], Marginal Fisher Analysis (MFA) [24], etc.

The existing discriminant analysis method design, however, is generally independent of classifier design. In other words, one

[*] Corresponding author.
E-mail addresses: csjyang@mail.njust.edu.cn (J. Yang), cslzhang@comp.polyu.edu.hk (L. Zhang), yangjy@mail.njust.edu.cn (J.-y. Yang), csdzhang@comp.polyu.edu.hk (D. Zhang).

0031-3203/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2011.01.009

does not consider what classifier would be used when trying to derive a discriminant analysis method. The derived discriminant analysis method, thus, can theoretically work with any classifier. The connection between discriminant analysis methods and classifiers is very loose. To generate an effective pattern recognition system, one needs to choose a classifier to match the designed discriminant analysis method well by experience. But, what classifier optimally matches the discriminant analysis method is generally unknown to us.

In this paper, we provide a way to design discriminant analysis methods that are bound with classifiers. We begin with a classifier and use it as a steerer to direct the design of a discriminator. Specifically we use the local mean based NN classification rule to direct the design of a discriminator, thus, the obtained discriminator, called local mean based NN discriminant analysis (LM-NNDA), matches the LM-NN classifier optimally. Therefore, the LM-NNDA based feature extractor and the LM-NN classifier can be seamlessly integrated into a pattern recognition system. In contrast to that LM-NNDA is a NN classifier induced discriminant analysis method, we further show that the classical FLDA is a minimum (Class-mean) distance classifier (or called nearest Class-mean classifier) induced discriminant analysis method. Therefore, FLDA is the most suitable feature extractor for the minimum distance classifier in theory.

Compared with the existing discriminant analysis methods, the most remarkable advantages of the proposed LM-NNDA method is its close connection to the NN classifier. Since the NN classifier has an asymptotical average error rate P satisfying $P^* \leq P \leq 2P^*$, where P^* is the Bayes error rate [36], it is reasonable to believe that the proposed LM-NNDA method can yield a Bayes suboptimal projection matrix which is connected to Bayes error via lower and upper bounds. In contrast, most existing discriminant analysis methods don't have this property.

In literature, we find that Hastie and Tibshirani's work [46] is quite interesting and related to ours. They proposed a method, coined discriminant adaptive nearest neighbor (DANN), to connect the discriminant analysis and nearest neighbor classification. The idea of DANN is quite different from ours, since DANN employs the between-class and within-class scatter information of a local linear discriminant analysis to define a new metric for computing neighborhoods in the K-NN classifier, whereas our method uses the decision rule of local mean based nearest neighbor classifier to derive a new discriminant analysis. In other words, the former uses the idea of discriminant analysis for the NN classifier design, while the later uses idea of the NN classifier for discriminant analysis design. Hastie and Tibshirani also presented a global dimensionality reduction (DR) method using the local between-class discriminant information [46]. As an extension of Hastie and Tibshirani's DANN method, Domeniconi et al. suggested a locally adaptive metric nearest-neighbor classification method by using the χ^2 distance for metric learning [47]. Bressan and Vitria showed a connection between nonparametric discriminant analysis and nearest neighbor classification [48]. Zhang et al. presented a discriminative nearest neighbor classification method by combining support vector machines and K-NN classifiers into one framework [49].

The remainder of this paper is organized as follows. Section 2 outlines the classical NN classifier and local mean based nearest neighbor (LM-NN) classifier. Section 3 develops the idea of the local mean based NN discriminant analysis (LM-NNDA) and the relevant algorithm. Section 4 reveals that FLDA is a minimum (Class-mean) distance classifier induced discriminant analysis method and the connection between LM-NNDA and FLDA. Section 5 describes the experimental methodology and results. Section 6 offers our conclusions and future work.

2. Outline of nearest neighbor classifiers

2.1. Nearest neighbor classifier

Suppose there are c known pattern classes. Let $\mathfrak{X}_i = \{\mathbf{X}_{ij} | j = 1,\ldots,M_i\}$ be the training sample set of Class i, where M_i is the number of training samples of Class i. For a given new sample $\mathbf{x}$, let us find its nearest neighbor $\mathbf{x}_{ir}$ in each class. $\mathbf{x}_{ir}$ is viewed as the prototype of Class i. The square distance from $\mathbf{x}$ to Class i is defined by

$$d_i(\mathbf{x}) = ||\mathbf{x} - \mathbf{x}_{ir}||^2. \tag{1}$$

Assume that the distance between $\mathbf{x}$ and Class l is minimal, i.e.

$$d_l(\mathbf{x}) = \min_i d_i(\mathbf{x}). \tag{2}$$

The decision rule of the 1-NN classifier is that $\mathbf{x}$ belongs to Class l.

Let $P_n(e)$ be the average probability of error for the 1-NN decision rule using n training samples. The asymptotical average error rate $P = \lim_{n\to\infty} P_n(e)$ satisfies the following property [36]:

$$P^* \leq P \leq P^*\Big(2 - \frac{c}{c-1}P^*\Big) < 2P^*, \tag{3}$$

where P^* is the Bayes error rate. Eq. (3) provides lower and upper bounds for the error rate of the 1-NN classifier in the case of an infinite number of samples.

The K-NN classifier naturally extends the idea of the 1-NN classifier by taking the K nearest neighbors and assigning the sign of the majority. Specifically, for a given test sample $\mathbf{x}$, suppose there are k_i samples belonging to Class i. If $k_l = \max_i k_i$,$\mathbf{x}$ belongs to Class l. The K-NN classifier has a similar asymptotical average error rate as shown in Eq. (3). However, it should be stressed that the NN classifier (either 1-NN or K-NN) requires a large number of training samples so as to approach the asymptotic performance. For a given limited number of training samples, the asymptotical error rate cannot be guaranteed.

2.2. Minimum (Class-mean) distance classifier

Minimum distance classifier uses the Class-mean as the prototype of the class, thus this classifier is also called the nearest Class-mean classifier. Let $\mathbf{m}_i$ $(i=1,\ldots,c)$ be the mean vector of the training samples in Class i. The square distance from $\mathbf{x}$ to Class i is defined by

$$d_i(\mathbf{x}) = ||\mathbf{x} - \mathbf{m}_i||^2. \tag{4}$$

If the distance between $\mathbf{x}$ and Class l is minimal, i.e. $d_l(\mathbf{x}) = \min_i d_i(\mathbf{x})$, the decision of the minimum distance (MD) classifier is that $\mathbf{x}$ belongs to Class l.

2.3. Local mean based nearest neighbor (LM-NN) classifier

Instead of searching for the 1-nearest neighbor of the given sample $\mathbf{x}$, the local mean NN classifier needs to find R-nearest neighbors of $\mathbf{x}$ from each class. Suppose the R-nearest neighbors of $\mathbf{x}$ in Class i are $\mathbf{x}_{ir}$, where $r=1,\ldots,R$. Let us calculate the mean vector of the these R-nearest neighbors $\mathbf{m}_i(\mathbf{x}) = (1/R)\sum_{r=1}^{R}\mathbf{x}_{ir}$, where $\mathbf{m}_i(\mathbf{x})$ is called the local mean of the sample $\mathbf{x}$ in Class i. $\mathbf{m}_i(\mathbf{x})$ is viewed as the prototype of Class i with respect to $\mathbf{x}$. The square distance from $\mathbf{x}$ to Class i is thus defined by

$$d_i(\mathbf{x}) = ||\mathbf{x} - \mathbf{m}_i(\mathbf{x})||^2. \tag{5}$$

If the distance between $\mathbf{x}$ and Class l is minimal, i.e. $d_l(\mathbf{x}) = \min_i d_i(\mathbf{x})$, we can make the decision that $\mathbf{x}$ belongs to Class l.

The LM-NN classifier can be thought of as a meaningful compromise between the minimum distance classifier and the nearest neighbor classifier. It has been demonstrated to be more powerful than the classical 1-NN and K-NN classifiers [8]. A possible reason is that the LM-NN classifier is more robust to outliers [8]. Actually even when there is no outlier in the training sample set, the LM-NN classifier can still be better. Fig. 1 provides an intuitive example for why the LM-NN classifier can outperform the 1-NN classifier. In the example, the two-class of samples are supposed to be linear separable and the number of given training samples is pretty small. Based on this small number of training samples, the 1-NN classifier produces a locally linear decision surface as shown in Fig. 1, which must lead to poor generalization performance. In such a case, the K-NN classifier encounters the same problem [25] as the 1-NN classifier and achieves an even worse decision surface. In contrast, the LM-NN classifier can yield a desired linear decision surface for this case. Note that here we use $R=2$ in the LM-NN classifier since the decision surface is one-dimensional (the nearest neighbor parameter R is chosen as $D+1$ for D-dimensional decision surface).

It is obvious that the local mean based nearest neighbor classifier is the nearest neighbor classifier when the nearest neighbor parameter $R=1$. The local mean based nearest neighbor classifier can be viewed as a generalized version of the classical nearest neighbor classifier. Therefore, we use the local mean NN decision rule in the design of our discriminant analysis method.

2.4. Parameter selection in the LM-NN classifier

The parameter R plays an important role in the performance of the LM-NN classifier, since it determines the degree of compromise between the MD classifier and the nearest neighbor classifier. If R is too small, for example reaches 1, the LM-NN classifier becomes the 1-NN classifier and loses the power of filling in "holes" caused by missing samples (just as shown in Fig. 1). If R is too large and approaching the training sample number of each class, the LM-NN classifier becomes the MD classifier and loses its power for dealing with nonlinearly separable problems. For instance, two classes of samples lie on or near the two concentric circles, respectively, as shown in Fig. 2, and suppose that for each class the mean of all 10 samples is exactly the center of circle. If $R=10$, the LM-NN classifier cannot make a decision because the Class-means share a same point.

From the theory of manifold learning [41,42,37,38], we can assume that each class of samples forms a lower-dimensional manifold (embedded in a high dimensional input space) which can reasonably be considered locally linear. For a given testing sample, it is reasonable to think that its R neighbors exist [21,25] on or near a local "flat patch" of the manifold, which is approximated by an $(R-1)$-dimensional local hyperplane spanned by R neighbors. From this point of view, it is reasonable to use the mean of the R neighbors, i.e. the centroid of the local hyperplane, to represent the local "flat patch". Conversely, from the viewpoint of manifolds, we have an insight into the LM-NN classifier itself. The parameter R, the number of the neighbors in the LM-NN classifier, should be chosen as $D+1$, where D is the local dimension, i.e. the dimension of the local "flat patch" of the manifold.

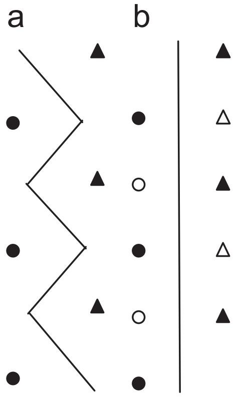

Fig. 1. Illustration of the decision surfaces of the nearest neighbor (NN) classifier and the local mean based nearest neighbor (LM-NN) classifier: (a) The decision surface of the NN classifier (it is locally linear due to the limited number of training samples) and (b) the decision surface of the LM-NN classifier. Note that in (b), the white circles and triangles represents the generated samples by the local mean operator ($R=2$). These samples help the LM-NN classifier produce a desired linear decision surface.

In the example as shown in Fig. 2, the 10 neighbors of one class are placed on a circle. The circle is viewed as a one-dimensional manifold, whose local patch is a line segment with a dimension of 1. In such a case, for a given test samples on or near the class manifold, we can find its R neighbors from each class, where R as is chosen as 2 rather than 10. Then, the local means of two classes are different and the LM-NN makes a right decision.

From this example, we can also find that there is a flexible scope allowing the parameter R to vary around $D+1$. When R varies from 1 to 3, the LM-NN can always achieve the right results. This provides us a flexibility to choose the parameter for achieving satisfying results. Our experimental results in Sections 5.1 and 5.2 further demonstrate this fact.

The parameter R can be theoretically determined by the local dimension of the manifold. However, evaluating the local dimension of a manifold accurately is very difficult or even impossible when there is a very limited number training samples available. Therefore, it is infeasible to choose R by evaluating the local dimension of the manifold. In practice, we generally choose R according to the number of training samples of each class. If there is a same number of training samples for each class, we always assume that there is a same local dimension shared by all classes and choose a proper, common R for each class which yields the best recognition performance. Otherwise, we choose $R_i=R_0M_i/M$, where M_i is the number of training samples in Class i, M is total number of the training samples across all classes, i.e. $M=\sum_{i=1}^{c}M_i$, and R_0 is a parameter which is shared by all classes. We use experiments to determine a proper R_0 for achieving satisfying recognition results.

3. Nearest neighbor rule induced discriminant analysis

This section will develop a discriminant analysis method under the guide of the LM-NN decision rule. The central idea of

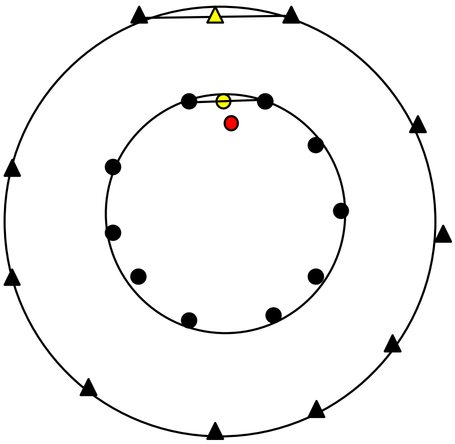

Fig. 2. Illustration of the choice of the parameter R in the LM-NN classifier for a nonlinearly separable problem. Here, the red point denotes a test sample, and the yellow points are local means belonging to different classes. An ideal parameter R is chosen as 2, since the local dimension of the class manifold is 1.

designing the method is to make the subsequent LM-NN classifier achieve the optimum performance in the reduced-dimensional space.

3.1. Basic idea

Let us first consider the problem in the observation space. Suppose there are c known pattern classes. Let $\mathfrak{X}=\{\mathbf{X}_{ij}\}$ be the training sample set, where $i=1,\ldots,c$ and $j=1,\ldots,M_i$. For each sample $\mathbf{x}_{ij}$, we can find its R-nearest neighbors in every class and calculate the corresponding local mean vector. Let $\mathbf{m}_{ij}^s$ be the local mean vector of $\mathbf{x}_{ij}$ in Class s. The distance between $\mathbf{x}_{ij}$ and Class s is

$$d_s(\mathbf{x}_{ij}) = ||\mathbf{x}_{ij}-\mathbf{m}_{ij}^s||^2. \tag{6}$$

To make the LM-NN classifier perform well on the training sample set, we want the within-class local distance $d_i(\mathbf{x}_{ij})$ as small as possible and the between-class local distance $d_s(\mathbf{x}_{ij})$ for each $s \neq i$ as large as possible. To this end, let us define the local within-class scatter of samples in the observation space as follows:

$$\frac{1}{M}\sum_{i,j} d_i(\mathbf{x}_{ij}) = \frac{1}{M}\sum_{i,j}\|\mathbf{x}_{ij}-\mathbf{m}_{ij}^i\|^2, \tag{7}$$

and the local between-class scatter of samples in the input space as follows:

$$\frac{1}{M(c-1)}\sum_{i,j}\sum_{s\neq i} d_s(\mathbf{x}_{ij}) = \frac{1}{M(c-1)}\sum_{i,j}\sum_{s\neq i}\|\mathbf{x}_{ij}-\mathbf{m}_{ij}^s\|^2, \tag{8}$$

where M is total number of training samples.

The local within-class scatter is actually the average of all pair of within-class local distances, and the local between-class scatter is the average of all pair of between-class local distances. According to LM-NN decision rule, larger local between-class scatter and smaller local within-class scatter will lead to better classification results in an average sense if samples are classified in the observation space.

Our goal is to find a linear discriminant transform

$$\mathbf{y}=\mathbf{P}^T\mathbf{x} \quad \text{where } \mathbf{P}=(\boldsymbol{\varphi}_1,\ldots,\boldsymbol{\varphi}_d), \tag{9}$$

such that the data points in the low-dimensional transformed space have the following properties:

(i) The local neighbor relationship is preserved.
(ii) The local between-class scatter of samples is maximized while at the same time the local within-class scatter of samples is minimized.

The first property is to guarantee that the R nearest neighbors of a point in the observation space are still the R nearest neighbors of the point in the transformed space. The second property aims to make the LM-NN classifier perform well in the transformed space.

3.2. Local mean based nearest neighbor discriminant analysis (LM-NNDA)

For simplicity, let first consider a one-dimensional linear transform $y=\boldsymbol{\varphi}^T\mathbf{x}$. Under this transform, each data point $\mathbf{x}_{ij}$ in observation space is mapped into $y_{ij}=\mathbf{P}^T\mathbf{x}_{ij}$ in a one-dimensional transformed space. Let the R-nearest neighbors of $\mathbf{x}_{ij}$ in Class s in the observation space be $\mathbf{x}_{sr}$, $r=1,\ldots,R$. Then, the local mean vector of $\mathbf{x}_{ij}$ in Class s in the observation space is $\mathbf{m}_{ij}^s=\sum_{r=1}^R \mathbf{x}_{sr}$. Since we assume the local neighbor relationship is preserved, in the transformed space, the R-nearest neighbors of the point y_{ij} in Class s is y_{sr}, $r=1,\ldots,R$. The local mean of y_{ij} in Class s in the transformed space is

$$\tilde{m}_{ij}^s = \sum_{r=1}^{K} y_{sr} = \sum_{r=1}^{K}\boldsymbol{\varphi}^T\mathbf{x}_{sr} = \boldsymbol{\varphi}^T\mathbf{m}_{ij}^s. \tag{10}$$

Now, let us define the local within-class scatter of samples in the transformed space as follows:

$$\frac{1}{M}\sum_{i,j} d_i(y_{ij}) = \frac{1}{M}\sum_{i,j}(y_{ij}-\tilde{m}_{ij}^i)^2 = \frac{1}{M}\sum_{i,j}(\boldsymbol{\varphi}^T\mathbf{x}_{ij}-\boldsymbol{\varphi}^T\mathbf{m}_{ij}^i)(\boldsymbol{\varphi}^T\mathbf{x}_{ij}-\boldsymbol{\varphi}^T\mathbf{m}_{ij}^i)^T$$

$$= \boldsymbol{\varphi}^T\left[\frac{1}{M}\sum_{i,j}(\mathbf{x}_{ij}-\mathbf{m}_{ij}^i)(\mathbf{x}_{ij}-\mathbf{m}_{ij}^i)^T\right]\boldsymbol{\varphi} = \boldsymbol{\varphi}^T\mathbf{S}_w^L\boldsymbol{\varphi},$$

where

$$\mathbf{S}_w^L = \frac{1}{M}\sum_{i,j}(\mathbf{x}_{ij}-\mathbf{m}_{ij}^i)(\mathbf{x}_{ij}-\mathbf{m}_{ij}^i)^T \tag{11}$$

is called the local within-class scatter matrix. It is easy to show that $\mathbf{S}_w^L$ is a nonnegative definite matrix.

Similarly let us define the local between-class scatter of samples in the transformed space as follows:

$$\frac{1}{M(c-1)}\sum_{i,j}\sum_{s\neq i} d_s(y_{ij}) = \frac{1}{M(c-1)}\sum_{i,j}\sum_{s\neq i}(y_{ij}-\tilde{m}_{ij}^s)^2$$

$$= \frac{1}{M(c-1)}\sum_{i,j}\sum_{s\neq i}[\boldsymbol{\varphi}^T(\mathbf{x}_{ij}-\mathbf{m}_{ij}^s)][\boldsymbol{\varphi}^T(\mathbf{x}_{ij}-\mathbf{m}_{ij}^s)]^T$$

$$= \boldsymbol{\varphi}^T\left[\frac{1}{M(c-1)}\sum_{i,j}\sum_{s\neq i}(\mathbf{x}_{ij}-\mathbf{m}_{ij}^s)(\mathbf{x}_{ij}-\mathbf{m}_{ij}^s)^T\right]\boldsymbol{\varphi} = \boldsymbol{\varphi}^T\mathbf{S}_b^L\boldsymbol{\varphi},$$

where

$$\mathbf{S}_b^L = \frac{1}{M(c-1)}\sum_{i,j}\sum_{s\neq i}(\mathbf{x}_{ij}-\mathbf{m}_{ij}^s)(\mathbf{x}_{ij}-\mathbf{m}_{ij}^s)^T \tag{12}$$

is called the local between-class scatter matrix. It is easy to show that $\mathbf{S}_b^L$ is a nonnegative definite matrix.

To maximize the between-class scatter and simultaneously to minimize the within-class scatter, we can choose to maximize the following criterion:

$$J(\boldsymbol{\varphi}) = \frac{\boldsymbol{\varphi}^T\mathbf{S}_b^L\boldsymbol{\varphi}}{\boldsymbol{\varphi}^T\mathbf{S}_w^L\boldsymbol{\varphi}}. \tag{13}$$

The optimal solution of the criterion in Eq. (13) is actually the generalized eigenvector $\boldsymbol{\varphi}$, of $\mathbf{S}_b^L\mathbf{X}=\lambda\mathbf{S}_w^L\mathbf{X}$ corresponding to the largest eigenvalue. Like FLDA, for multiple-class problems, one projection axis $\boldsymbol{\varphi}$ is not enough for discrimination, so we generally need to find a set of projection axes. Similar to the way adopted by FLDA to get multiple projection axes, we can calculate the generalized eigenvectors $\boldsymbol{\varphi}_1,\ldots,\boldsymbol{\varphi}_d$ of $\mathbf{S}_b^L\mathbf{X}=\lambda\mathbf{S}_w^L\mathbf{X}$ corresponding to the d largest eigenvalues and use them as projection axes to produce a transform matrix $\mathbf{P}=(\boldsymbol{\varphi}_1,\ldots,\boldsymbol{\varphi}_d)$, where d is the number of projection axes chosen. The linear transformation $\mathbf{y}=\mathbf{P}^T\mathbf{x}$ forms a feature extractor which reduces the dimension of original feature vectors to d.

In summary of the description above, the local mean based nearest neighbor discriminant analysis (LM-NNDA) algorithm is given below:

3.2.1. The LM-NNDA algorithm

Step 1: For each sample point $\mathbf{x}_{ij}$, find its R-nearest neighbors in every class and calculate the corresponding local mean vector. Let $\mathbf{m}_{ij}^s$ be the local mean vector of $\mathbf{x}_{ij}$ in Class s.

Step 2: Construct the local within-class scatter matrix $\mathbf{S}_w^L$ and the local between-class scatter matrix $\mathbf{S}_b^L$ using Eqs. (11) and (12). Calculate the generalized eigenvectors $\boldsymbol{\varphi}_1,\ldots,\boldsymbol{\varphi}_d$ of $\mathbf{S}_b^L$ and $\mathbf{S}_w^L$

corresponding to the d largest generalized eigenvalues. Let $\mathbf{P}=(\boldsymbol{\varphi}_1,\ldots,\boldsymbol{\varphi}_d)$.

Step 3: For a given sample $\mathbf{x}$, its feature vector $\mathbf{y}$ is obtained by the linear transform $\mathbf{y}=\mathbf{P}^T\mathbf{x}$.

It should be noted that unlike FLDA, which can generate at most $c-1$ effective projection axes, the LM-NNDA algorithm can yield more than $c-1$ effective projection axes, since the rank of $\mathbf{S}_b^L$ is generally much larger than $c-1$.

In addition, since LM-NNDA involves a parameter R, how to choose the parameter is a first problem. Here, we assume that a common R is shared by all classes and determine a proper R which yields the best recognition performance by experimental evaluation, just as the way used in the LM-NN classifier. It should be mentioned that our experiments show that LM-NNDA is insensitive to the variation of R, that is, the performance of LM-NNDA is steady when R ranges in a relative large interval.

Finally we would like to analyze the computational complexity of LM-NNDA. In the construction of the between-class and within-class scatter matrices $\mathbf{S}_b^L$ and $\mathbf{S}_w^L$, for each training sample, we need to find its R nearest neighbors within each class. Therefore, compared to the FLDA method, an additional computational cost of LM-NNDA is required for the nearest neighbor search. The naive (linear) search of the R neighbors of one point within Class i has a running time of $O(RM_iD)$, where M_i is the number of samples in the Class i and D is of dimension of the pattern vectors. So the computational complexity for nearest neighbor search in LM-NNDA is $O(RM^2D)$, where is M is total number of training samples, $M=\sum_{i=1}^{c}M_i$. The naive search algorithm only suits for small sample size cases. For large sample size cases, more advanced nearest neighbor search algorithms with lower computational complexity can be used instead [43].

3.3. Implementation of LM-NNDA in small sample size cases

In the small sample size cases where the number of training samples is smaller than the dimension of the image vector space, the local within-class scatter matrix $\mathbf{S}_w^L$ is always singular because the following proposition holds:

Proposition 1. *The rank of the local within-class scatter matrix* $\mathbf{S}_w^L$ *is equal or less than* $M-c$, i.e. $\text{rank}(\mathbf{S}_w^L)\leq M-c$, *where M is number of training samples and c is the number of classes.*

Proof. First of all, let us rewrite $\mathbf{S}_{\mathrm{w}}^{\mathrm{L}}=(1/M)\sum_{i=1}^{c}\mathbf{S}_w^{L_i}$, where $\mathbf{S}_w^{L_i}=\sum_{j=1}^{M_i}(\mathbf{x}_{ij}-\mathbf{m}_{ij}^i)(\mathbf{x}_{ij}-\mathbf{m}_{ij}^i)^T$.

$\mathbf{S}_w^{L_i}$ can be viewed as the local scatter matrix of Class i. For convenience of discussion, we would like to express $\mathbf{S}_w^{L_i}$ in a matrix form. To this end, let us define a M_i-dimensional column vector $\mathbf{H}_{ij}=(H_{ij}^k)_{M_i\times 1}$ for each sample point $\mathbf{x}_{ij}$, whose kth element is given below:

$$H_{ij}^k=\begin{cases}-\frac{1}{K} & \text{if } \mathbf{x}_{ik} \text{ is among } K \text{ nearest nieghbors of } \mathbf{x}_{ij},\\ 1 & \text{if } k=j,\\ 0 & \text{otherwise}\end{cases}.$$

Letting $\mathbf{X}_i=[\mathbf{x}_{i1},\ldots,\mathbf{x}_{iM_i}]$ and $\mathbf{H}_i=[\mathbf{H}_{i1},\ldots,\mathbf{H}_{iM_i}]$, we have

$$\mathbf{S}_w^{L_i}=\sum_{j=1}^{M_i}(\mathbf{X}_i\mathbf{H}_{ij})(\mathbf{X}_i\mathbf{H}_{ij})^T=\mathbf{X}_i\left(\sum_{j=1}^{M_i}\mathbf{H}_{ij}\mathbf{H}_{ij}^T\right)\mathbf{X}_i^T=\mathbf{X}_i(\mathbf{H}_i\mathbf{H}_i^T)\mathbf{X}_i^T.$$

Now, let us consider the rank of the M_i by M_i matrix $\mathbf{H}_i=[\mathbf{H}_{i1},\ldots,\mathbf{H}_{iM_i}]$. Since the sum of the all elements in $\mathbf{H}_{ij}$ is zero, adding all but the first rows of the matrix $\mathbf{H}_i$ to the first row, we get the following matrix:

$$\begin{bmatrix}0 & 0 & \cdots & 0\\ H_{i1}^2 & 1 & \cdots & H_{iM_i}^2\\ \vdots & \vdots & \ddots & \vdots\\ H_{i1}^{M_i} & H_{i2}^{M_i} & \cdots & 1\end{bmatrix}$$

Thus, $\text{rank}(\mathbf{H}_i)\leq M_i-1$. From the singular value decomposition (SVD) theorem [26], we know that $\mathbf{H}_i$ and $\mathbf{H}_i\mathbf{H}_i^T$ have the same rank. Therefore, $\text{rank}(\mathbf{H}_i\mathbf{H}_i^T)\leq M_i-1$.

Let $\mathbf{X}=[\mathbf{X}_1,\ldots,\mathbf{X}_c]$ and $\mathbf{H}=\text{diag}(\mathbf{H}_1\mathbf{H}_1^T,\ldots,\mathbf{H}_c\mathbf{H}_c^T)$. It is easy to derive that

$$\text{rank}(\mathbf{H})\leq\sum_{i=1}^{c}(M_i-1)=M-c.$$

Then, we have

$$\mathbf{S}_w^L=\frac{1}{M}\sum_{i=1}^{c}\mathbf{S}_w^{L_i}=\frac{1}{M}\sum_{i=1}^{c}\mathbf{X}_i(\mathbf{H}_i\mathbf{H}_i^T)\mathbf{X}_i^T=\frac{1}{M}\mathbf{X}\mathbf{H}\mathbf{X}^T.$$

Therefore $\text{rank}(\mathbf{S}_w^L)\leq M-c$. □

In real-world applications, the given M training samples are generally linear independent in the high-dimensional input space. The rank of $\mathbf{S}_w^L$ is usually $M-c$. To avoid overfitting in small sample size cases, we borrow the idea in [9,10] and use PCA to reduce the dimension of the input space such that $\mathbf{S}_w^L$ is nonsingular in the PCA-transformed space. We then perform LM-NNDA based on PCA-transformed features. To further enhance the robustness of the LM-NNDA algorithm, we use the following technique to regularize the local within-class scatter matrix $\mathbf{S}_w^L$:

$$\mathbf{S}_w^L\leftarrow\mathbf{S}_w^L+\alpha\mathbf{I},\tag{14}$$

where $\mathbf{I}$ is the identity matrix and α is chosen as $\alpha=0.001\,\text{trace}(\mathbf{S}_{\mathrm{w}}^{\mathrm{L}})$ in this paper.

3.4. A special case of LM-NNDA: NNDA

Particularly, when the nearest neighbor parameter $R=1$, LM-NNDA becomes a nearest neighbor rule induced discriminant analysis (NNDA). In the training process of NNDA, for each sample point $\mathbf{x}_{ij}$, we need to find its within-class nearest neighbor and all between-class nearest neighbors. Then, we try to minimize the average distance between every point and its within-class nearest neighbor and simultaneously to maximize the average distance between every point and its between-class nearest neighbor.

A NNDA closely related dicriminant analysis is the nonparametric margin maximum criterion (NMMC) method proposed by Qiu and Wu [27]. The basic idea of NMMC is to find the within-class *furthest* neighbor (rather than the within-class *nearest* neighbor) and the between-class nearest neighbor of each sample point, and then to minimize the average distance between every point and its within-class *furthest* neighbor and simultaneously to maximize the average distance between every point and its between-class nearest neighbor. The most remarkable difference between NMMC and NNDA is that the former focuses on the within-class *furthest* neighbor of a sample while the later focuses on the within-class *nearest* neighbor. Focusing on the within-class *furthest* neighbor, however, may encounter the following problems:

(i) The within-class *furthest* neighbors are more likely to be outliers, so depending on them to characterize the within-class scatter is not very robust.

(ii) In some cases, minimizing the distance between a point and its within-class *furthest* neighbor does not make sense for classification, as shown in Fig. 3. Reducing the distance between a sample X_1 and its within-class furthest neighbor X_2 has no effect on the classification of the two-class samples.

In contrast, the NNDA method does not have the foregoing problems since it uses the within-class *nearest* neighbors to characterize the within-class scatter. This within-class *nearest* neighbors based within-class scatter characterization plus the between-class *nearest* neighbors based between-class scatter characterization make NNDA more suitable for the NN classifier than NMMC.

4. Further discussion

This section shows that FLDA is a minimum distance classification rule induced discriminant analysis method, and then reveals the connection between LM-NNDA and FLDA. Finally we elucidate the dependency of LM-NNDA and the LM-NN classifier in a special case.

4.1. FLDA

FLDA seeks to find a projection axis such that the Fisher criterion (i.e. the ratio of the *between-class scatter* to the *within-class scatter*) is maximized after the projection of samples. The between-class and within-class scatter matrices $\mathbf{S}_b$ and $\mathbf{S}_w$ are defined by

$$\mathbf{S}_b = \frac{1}{M}\sum_{i=1}^{c} M_i(\mathbf{m}_i - \mathbf{m}_0)(\mathbf{m}_i - \mathbf{m}_0)^{\mathrm{T}}, \tag{15}$$

$$\mathbf{S}_w = \frac{1}{M}\sum_{i=1}^{c}\sum_{j=1}^{M_i}(\mathbf{x}_{ij} - \mathbf{m}_i)(\mathbf{x}_{ij} - \mathbf{m}_i)^{\mathrm{T}}, \tag{16}$$

where $\mathbf{x}_{ij}$ denotes the jth training sample in class I, c is the number of classes, M_i is the number of training samples in class I, $\mathbf{m}_i$ is the mean of the training samples in class I, $\mathbf{m}_0$ is the total mean of training samples, i.e. $\mathbf{m}_0 = (1/M)\sum_{j=1}^{M}\mathbf{x}_j = (1/M)\sum_{i=1}^{c} M_i\mathbf{m}_i$. Specially when each class has the same number of training samples, the between-class scatter matrix becomes

$$\mathbf{S}_b = \frac{1}{c}\sum_{i=1}^{c}(\mathbf{m}_i - \mathbf{m}_0)(\mathbf{m}_i - \mathbf{m}_0)^{\mathrm{T}}, \tag{17}$$

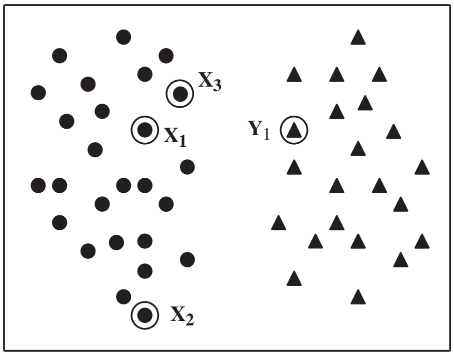

Fig. 3. Illustration of the within-class furthest neighbor, the within-class nearest neighbor and the between-class nearest neighbor of a sample point. For the sample point X_1, its within-class furthest neighbor is X_2, its within-class nearest neighbor is X_3, and it is between-class nearest neighbor is Y_1. It is obvious that reducing the distance between X_1 and its within-class furthest neighbor X_2 does not make sense for the classification of the two-class problem.

The Fisher criterion is defined by

$$J_F(\boldsymbol{\varphi}) = \frac{\boldsymbol{\varphi}^T\mathbf{S}_b\boldsymbol{\varphi}}{\boldsymbol{\varphi}^T\mathbf{S}_w\boldsymbol{\varphi}}. \tag{18}$$

The stationary points of $J_F(\boldsymbol{\varphi})$ are the generalized eigenvectors $\boldsymbol{\varphi}_1,\boldsymbol{\varphi}_2,\ldots,\boldsymbol{\varphi}_d$, of $\mathbf{S}_b\mathbf{X}=\lambda\,\mathbf{S}_w\mathbf{X}$ corresponding to d largest eigenvalues. These stationary points form the coordinate system of FLDA.

4.2. FLDA is a minimum distance classification rule induced discriminant analysis

In this section, we first develop a MD classification rule induced discriminant analysis method, and then show its equivalence to FLDA when each class has the same number of training samples.

Let us consider the problem in the $y=\boldsymbol{\varphi}^T\mathbf{x}$ transformed space. Each data point $\mathbf{x}_{ij}$ in observation space is mapped into $y_{ij}=\boldsymbol{\varphi}^T\mathbf{x}_{ij}$ in the transformed space. The class-mean vector $\mathbf{m}_i$ ($i=1,\ldots,c$) and the total mean vector $\mathbf{m}_0$ in observation space is thus mapped into $\tilde{m}_i = \boldsymbol{\varphi}^T\mathbf{m}_i$ and $\tilde{m}_0 = \boldsymbol{\varphi}^T\mathbf{m}_0$, respectively, in the transformed space. To make the MD classifier perform well, we try to minimize the following within-class scatter:

$$\begin{aligned}\frac{1}{M}\sum_{i,j} d_i(y_{ij}) &= \frac{1}{M}\sum_{i,j}(y_{ij} - \tilde{m}_i)^2 = \frac{1}{M}\sum_{i,j}[\boldsymbol{\varphi}^T(\mathbf{x}_{ij} - \mathbf{m}_i)][\boldsymbol{\varphi}^T(\mathbf{x}_{ij} - \mathbf{m}_i)]^T \\ &= \boldsymbol{\varphi}^T\mathbf{S}_w\boldsymbol{\varphi},\end{aligned} \tag{19}$$

and simultaneously to maximize the following between-class scatter:

$$\begin{aligned}\frac{1}{M(c-1)}\sum_{i,j}\sum_{s\neq i} d_s(y_{ij}) &= \frac{1}{M(c-1)}\sum_{i,j}\sum_{s\neq i}(y_{ij} - \tilde{m}_s)^2 \\ &= \frac{1}{M(c-1)}\sum_{i,j}\sum_{s\neq i}[(y_{ij} - \tilde{m}_i) + (\tilde{m}_i - \tilde{m}_s)]^2 \\ &= \frac{1}{M(c-1)}\sum_{i,j}\sum_{s\neq i}(y_{ij} - \tilde{m}_i)^2 + \frac{1}{M(c-1)}\sum_{i,j}\sum_{s\neq i}(\tilde{m}_i - \tilde{m}_s)^2 \\ &\quad + \frac{2}{M(c-1)}\sum_{i,j}\sum_{s\neq i}(y_{ij} - \tilde{m}_i)(\tilde{m}_i - \tilde{m}_s) \\ &= \frac{1}{M}\sum_{i,j}(y_{ij} - \tilde{m}_i)^2 + \frac{1}{M(c-1)}\sum_{i}\sum_{s} M_i(\tilde{m}_i - \tilde{m}_s)^2 + 0 \\ &= \boldsymbol{\varphi}^T\mathbf{S}_w\boldsymbol{\varphi} + \frac{1}{M(c-1)}\sum_{i}\sum_{s} M_i(\tilde{m}_i - \tilde{m}_0 + \tilde{m}_0 - \tilde{m}_s)^2 \\ &= \boldsymbol{\varphi}^T\mathbf{S}_w\boldsymbol{\varphi} + \frac{1}{M(c-1)}\left[c\sum_{i} M_i(\tilde{m}_i - \tilde{m}_0)^2 + \sum_{i} M_i\sum_{s}(\tilde{m}_s - \tilde{m}_0)^2 + 0\right] \\ &= \boldsymbol{\varphi}^T\mathbf{S}_w\boldsymbol{\varphi} + \frac{1}{M(c-1)}\left[c\sum_{i} M_i(\tilde{m}_i - \tilde{m}_0)^2 + M\sum_{s}(\tilde{m}_s - \tilde{m}_0)^2\right] \\ &= \boldsymbol{\varphi}^T\mathbf{S}_w\boldsymbol{\varphi} + \frac{1}{M(c-1)}\sum_{i}(cM_i + M)(\boldsymbol{\varphi}^T\mathbf{m}_i - \boldsymbol{\varphi}^T\mathbf{m}_0)^2 \\ &= \boldsymbol{\varphi}^T\mathbf{S}_w\boldsymbol{\varphi} + \boldsymbol{\varphi}^T\left[\frac{1}{M(c-1)}\sum_{i}(cM_i + M)(\mathbf{m}_i - \mathbf{m}_0)(\mathbf{m}_i - \mathbf{m}_0)^T\right]\boldsymbol{\varphi}.\end{aligned} \tag{20}$$

When each class has the same number of training samples, i.e. $M_i=M/c$ ($i=1,\ldots,c$), the foregoing between-class scatter becomes

$$\begin{aligned}\frac{1}{M(c-1)}\sum_{i,j}\sum_{s\neq i} d_s(y_{ij}) &= \boldsymbol{\varphi}^T\mathbf{S}_w\boldsymbol{\varphi} + \boldsymbol{\varphi}^T\left[\tfrac{2}{c-1}\sum_{i}(\mathbf{m}_i - \mathbf{m}_0)(\mathbf{m}_i - \mathbf{m}_0)^T\right]\boldsymbol{\varphi} \\ &= \boldsymbol{\varphi}^T\mathbf{S}_w\boldsymbol{\varphi} + \tfrac{2}{c-1}\boldsymbol{\varphi}^T\mathbf{S}_b\boldsymbol{\varphi}.\end{aligned} \tag{21}$$

In such a case, the criterion of the MD classification rule induced discriminant analysis is given by

$$J_M(\boldsymbol{\varphi}) = \frac{\boldsymbol{\varphi}^T\mathbf{S}_w\boldsymbol{\varphi}+(2/c-1)\boldsymbol{\varphi}^T\mathbf{S}_b\boldsymbol{\varphi}}{\boldsymbol{\varphi}^T\mathbf{S}_w\boldsymbol{\varphi}} \tag{22}$$

It is easy to show that the following equivalent relationships hold:

$$J_M(\boldsymbol{\varphi}) \Leftrightarrow \frac{(2/c-1)\boldsymbol{\varphi}^T\mathbf{S}_b\boldsymbol{\varphi}}{\boldsymbol{\varphi}^T\mathbf{S}_w\boldsymbol{\varphi}} \Leftrightarrow \frac{\boldsymbol{\varphi}^T\mathbf{S}_b\boldsymbol{\varphi}}{\boldsymbol{\varphi}^T\mathbf{S}_w\boldsymbol{\varphi}} = J_F(\boldsymbol{\varphi}). \tag{23}$$

This equivalence means that the MD classification rule induced discriminant analysis method has the same solution with FLDA when each class has the same number of training samples. Therefore, FLDA can be viewed as a MD classification rule induced discriminant analysis method. From this point of view, it can be said that FLDA is the most suitable feature extractor for the minimum distance classifier.

4.3. Connecting LM-NNDA to FLDA

In this section, we will show that, when each class has the same number of training samples, LM-NNDA is approaching to FLDA with the increase of the nearest neighbor parameter R.

Suppose that the training sample number of each class is $M_i=M/c$ ($i=1,\ldots,c$). Let $\mathbf{m}_{ij}^s$ be the local mean vector of $\mathbf{x}_{ij}$ in Class s. It is easy to know that $\mathbf{m}_{ij}^s$ is approaching to the mean of Class s, $\mathbf{m}_s$, when R approaches to M_i, i.e.

$$\mathbf{m}_{ij}^s \to \mathbf{m}_s \quad \text{when} \quad R \to M_i. \tag{24}$$

Therefore, LM-NNDA approaches to the minimum distance classification rule induced discriminant analysis method described in the foregoing section when R approaches to M_i. Since the minimum distance classification rule induced discriminant analysis has been proven equivalent to FLDA, we can conclude that LM-NNDA approaches to FLDA when R approaches to the class training sample number M_i. This is interestingly consistent with the fact that the LM-NN classifier approaches to the MD classifier when R approaches to the class training sample number M_i, noticing that FLDA is a MD classification rule induced discriminant analysis method.

4.4. Dependency of LM-NNDA and the LM-NN classifier

The design of the LM-NNDA method is intuitively based the decision rule of the LM-NN classifier. That is, in order to make the subsequent LM-NN classifier achieve a good performance, we model LM-NNDA by maximizing the average point-to-intra-class distance and simultaneously minimizing the average point-to-inter-class distance. Thus, we conclude that LM-NNDA is the most suitable feature extraction method for the LM-NN classifier. In this section, we try to provide some theoretical analysis on the dependency of LM-NNDA and the LM-NN classifier. We will discuss in what cases, LM-NNDA is guaranteed to be the statistically optimal for the LM-NN classifier.

Let us first begin with the FLDA method, a special case of LM-NNDA. It is known that the transform matrix of FLDA is the Bayes optimal when each class of samples shares a normal distribution with an identical covariance matrix [34]. In this case, the within-class scatter matrix of FLDA, $\mathbf{S}_w=\mathbf{S}_w^i$, for $i=1,\ldots,c$, where $\mathbf{S}_w^i$ is the covariance matrix of Class i. Then, in such a case, we can ensure that the transformed samples of each class share a normal distribution and the covariance matrix is an identity matrix because the FLDA algorithm can lead to

$$\mathbf{P}^T\mathbf{S}_w\mathbf{P}=\mathbf{I} \quad \text{where} \quad \mathbf{P}=(\boldsymbol{\varphi}_1,\boldsymbol{\varphi}_2,\ldots,\boldsymbol{\varphi}_d) \quad \text{where } \mathbf{I} \text{ is an identity matrix.} \tag{25}$$

The mean minimum distance classifier is the Bayes optimal when each class of samples has a normal distribution with an identity covariance matrix. From the above analysis, we can now conclude that FLDA is the statistically optimal for the MD classifier in the case that each class of samples shares a normal distribution with a same covariance matrix.

The above conclusion on FLDA can be extended to LM-NNDA because LM-NNDA can be viewed as a "locally" FLDA method. Specifically let us look at class manifolds and focus on the set of R samples within a local patch, which can be thought of as a subclass. It is easy to show that LM-NNDA is a subclass based FLDA method, following the derivation process given in Section 4.2. If the all subclasses share a normal distribution with an identity covariance matrix, the subclass based FLDA method (i.e. LM-NNDA) is the statistically optimal for the subclass based MD classifier. Notice that the subclass (formed by the R local samples) based MD classifier is the LM-NN classifier. Therefore, we can conclude that when the class manifolds have the same local distribution, i.e. all set of R local samples (subclasses) share a normal distribution with an identical covariance matrix, LM-NNDA is the statistically optimal for the LM-NN classifier.

It should be noted that here we only elucidate the dependency of LM-NNDA and the LM-NN classifier in a special case. For more general cases, statistical analysis on the dependency of LM-NNDA and the LM-NN classifier is still an open problem.

4.5. Connecting LM-NNDA to LB-LDA

Hastie and Tibshirani presented a dimensionality reduction method using the local discriminant information [46]. Their method seeks the subspace spanned by eigenvectors of the average local between sum-of-squares matrices based on the globally sphered data. In other words, the method characterizes the between-class scatter information locally, whereas characterizes the within-class scatter information globally. Therefore, Hastie and Tibshirani's method is essentially a semi-local dimensionality reduction method, which is named local between-class linear discriminant analysis (LB-LDA) in this paper. In contrast, LM-NNDA is a full-local dimensionality reduction method, which characterizes both the between-class scatter information and the within-class scatter information locally. Specifically LM-NNDA use the local within-class scatter matrix as shown in Eq. (11), while LB-LDA uses the global within-class scatter matrix as LDA as shown in Eq. (16).

Moreover, as far as local between-class scatter information is concerned, LM-NNDA and LB-LDA have different characterizations. Specifically, for each sample point $\mathbf{x}_{ij}$, suppose $\mathbf{m}_{ij}^s$ is the local mean vector of $\mathbf{x}_{ij}$ in Class s. LM-NNDA uses $\mathbf{x}_{ij}$ and its between-class local means $\mathbf{m}_{ij}^s$ ($s \neq i$) to describe scatters and to construct the local between-class scatter matrix as shown in Eq. (12), while LB-LDA uses the within-class local mean of $\mathbf{x}_{ij}$, $\mathbf{m}_{ij}=\mathbf{m}_{ij}^i$, and the overall local mean $\overline{\mathbf{m}}_{ij}=(1/c)\sum_{s=1}^{c}\mathbf{m}_{ij}^s$ to describe scatters and to construct the local between-class scatter matrix as follows:

$$\mathbf{S}_b^{LB} = \sum_{i,j}(\mathbf{m}_{ij}-\overline{\mathbf{m}}_{ij})(\mathbf{m}_{ij}-\overline{\mathbf{m}}_{ij})^T \tag{26}$$

It is evident that the formulation in Eq. (26) is different from that in Eq. (12).

Finally it should be stressed that based on the characterization of the local between-class scatter matrix $\mathbf{S}_b^{LB}$ and the global within-class scatter matrix $\mathbf{S}_w$, LB-LDA does not show direct connections to classifiers. In contrast, the characterization of the local between-class scatter matrix $\mathbf{S}_b^L$ and the local within-class scatter matrix $\mathbf{S}_w^L$ in LM-NNDA guarantees the connections between LM-NNDA and local mean based nearest neighbor classifier.

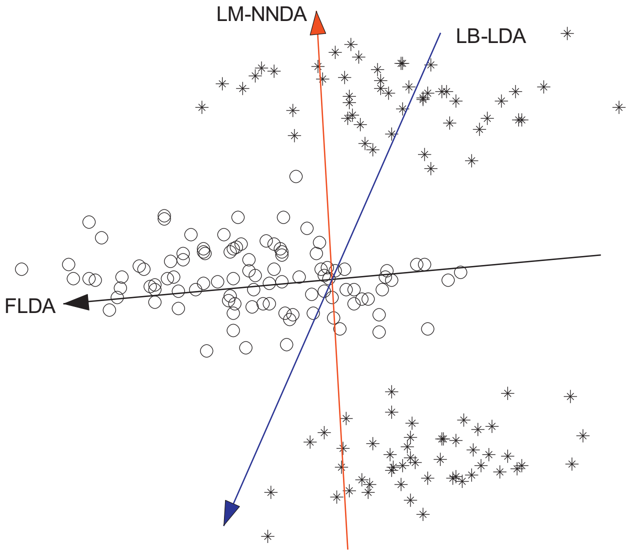

Fig. 4. Illustration of the projection direction of LM-NNDA (red), LB-LDA (blue) and FLDA (black) for two classes of samples (for interpretation of the references to color in this figure legend, the reader is referred to the web version of this article).

Fig. 4 shows an illustrative example where LM-NNDA yields different projection direction as opposed to LB-LDA and FLDA. In this example, there are two classes of samples. The samples of one class follows a normal distribution, while the samples of the other class form two clusters, each following a normal distribution. In the figure, the red line represents the projection direction of LM-NNDA, the blue line represents that of LB-LDA, and the black line represents that of FLDA. It appears that LM-NNDA produces a more meaningful projection direction than LB-LDA and FLDA for separating the two classes of samples. After being projected onto the direction of LM-NNDA, samples can be well classified under a local mean based nearest neighbor classifier. LB-LDA produces a projection direction somewhat close to LM-NNDA, but still causes some overlapping of the two classes of samples after the projection. Note that LB-LDA uses the global within-class scatter matrix to sphere data. In our opinion, it is the global within-class scatter matrix that misleads the projection direction of LB-LDA. When one class contains two clusters of samples as shown in Fig. 4, the global within-class scatter matrix fails to characterize the structure of data well. In contrast, the local within-class scatter matrix can provide a much better characterization. Therefore, LM-NNDA can yield a desirable projection direction.

5. Experiments

In this section, the local mean based nearest neighbor discriminant analysis (LM-NNDA) method is evaluated using the CENPARMI handwritten numeral database, the NUST603 handwritten Chinese character database, the ETH80 object category database, and the FERET face image database and compared with the PCA, FLDA, Locality Preserving Projection (LPP) [22], and the nonparametric margin maximum criterion (NMMC) method [27].

It is natural that the proposed LM-NNDA based feature extractor and the LM-NN classifier can be integrated into a complete pattern recognition system. In this system, however, the neighbor parameter R can be chosen differently in the feature extractor and classifier. The value of R is determined by the local dimension of the pattern class manifold. Since the dimension of the original pattern space is larger than the transformed feature space, the value of R in LM-NNDA (which is performed in the original pattern space) is generally chosen larger than the value of R in the LM-NN classifier. For distinction, we use R_1 to denote the parameter in LM-NNDA and R_2 in the LM-NN classifier in the following experiments.

5.1. Experiment using the CENPARMI handwritten numeral database

The experiment was conducted on Concordia University CENPARMI handwritten numeral database. The database contains 6000 samples of 10 numeral classes (each class has 600 samples). The original 121-dimensional Legendre moment features [28] were extracted for each sample and used here. In our experiment, the first 200 samples of each class are used to compose the training set, the second 200 samples of each class compose the validation set, and the remaining 200 samples form the test set.

In order to provide a baseline, we first apply the three classifiers, the local-mean based nearest neighbor (LM-NN) classifier, the K nearest neighbor (K-NN) classifier and the minimum (class-mean) distance (MD) classifier, to the original 121-dimensional Legendre moment features. The classification results on the validation set are shown in Fig. 5. It appears that the LM-NN classifier consistently outperforms the K-NN classifier. The performance of the K-NN classifier does not improve with the increase of K for this database. In contrast, the LM-NN classifier is more robust with the variation of the parameter. From these results, we choose the optimal parameter $K=1$ for the K-NN classifier and $K=6$ for the LM-NN classifier. Based on these parameters, we obtain the recognition results of both classifiers on the test set, as shown in Table 1. Table 1 shows us that the LM-NN classifier performs better than the K-NN and MD classifiers.

PCA, FLDA, LB-LDA [46], LPP, and the proposed LM-NNDA are, respectively, used for feature extraction based on the original Legendre moment features. For each of the five methods, we use the three classifiers mentioned above. Taking the combination of LM-NNDA and the LM-NN classifier as an example, we show how to tune the parameters R_1 in the feature extractor and R_2 in the classifier on the validation set. Let R_1 vary from 5 to 40 with an interval of 5, R_2 varies from 2 to 20 with an interval of 2, and the dimension of the extracted features vary from 2 to 50 with an interval of 2. The maximal classification rates over the variation of dimensions corresponding to R_1 and R_2 are shown in Table 2. From Table 2, we can see that the performance of LM-NNDA is very robust with the variation of the parameter; it remains above 95.6% despite the variation of R_1 and R_2 within the given range.

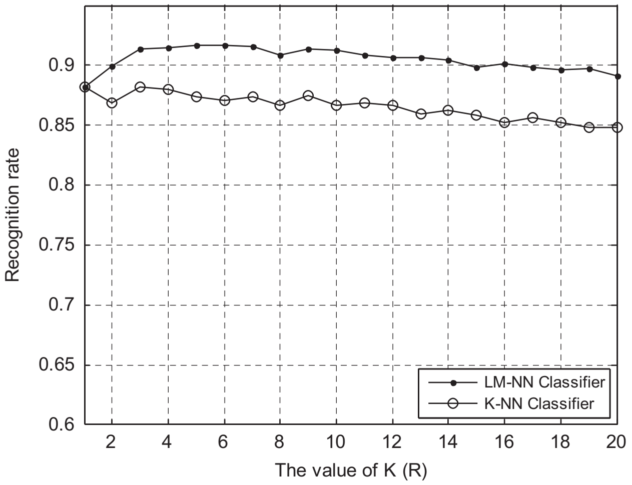

Fig. 5. The recognition rates of the LM-NN classifier and the K-NN classifier over the variation of parameters on the validation set of the CENPARMI database.

Table 1
The recognition rates (%) of the three different classifiers based on the original 121-dimensional Legendre moment features on the test set.

MD	K-NN (K=1)	LM-NN (R=6)
54.4	88.2	91.8

Table 2
The maximal classification rates over the variation of dimensions corresponding to R_1 in LM-NNDA and R_2 in the LM-NN classifier on the validation set.

	R_2									
R_1	2	4	6	8	10	12	14	16	18	20
5	0.960	0.969	0.965	0.965	0.963	0.959	0.958	0.958	0.957	0.958
10	0.959	0.966	0.969	0.970	0.963	0.961	0.961	0.957	0.958	0.956
15	0.961	0.966	0.971	0.969	0.965	0.961	0.960	0.960	0.958	0.958
20	0.958	0.966	*0.973*	0.968	0.967	0.963	0.960	0.959	0.959	0.958
25	0.959	0.967	*0.973*	0.969	0.968	0.962	0.961	0.960	0.959	0.958
30	0.959	0.968	0.972	0.969	0.968	0.962	0.962	0.960	0.960	0.958
35	0.957	0.968	0.971	0.971	0.969	0.963	0.962	0.961	0.960	0.959
40	0.956	0.969	0.971	0.972	0.969	0.965	0.963	0.962	0.962	0.960

This provides us enough flexibility for parameter selection in LM-NNDA. Here the optimal parameter combination is chosen as R_1=20 and R_2=6, which correspond to an optimal dimension d=48. In a similar way, we can find optimal parameters R_1 and R_2 for different combinations of other feature extraction methods and classifiers. Based on these parameters, the recognition rate of each method on the test set versus the variation of dimensions is shown in Fig. 6.

Fig. 6(a) and (b) shows that the LM-NNDA with the LM-NN and NN classifiers noticeably outperforms PCA, LB-LDA and LPP with the same classifiers, irrespective of the variation in dimensions. FLDA can only extract $c-1=9$ features in this experiment since there are totally 10 classes. Although the nine FLDA features are as effective as the first nine LM-NNDA features, this small number of features is obviously not enough to represent numeral pattern for recognition purposes. Fig. 6(c), however, shows that FLDA consistently outperforms the other four feature extraction method in terms of the MD classifier. The above results are completely consistent with our analysis in the foregoing sections, that is, LM-NNDA is the LM-NN classification rule induced discriminant analysis, while FLDA is the MD classification rule induced.

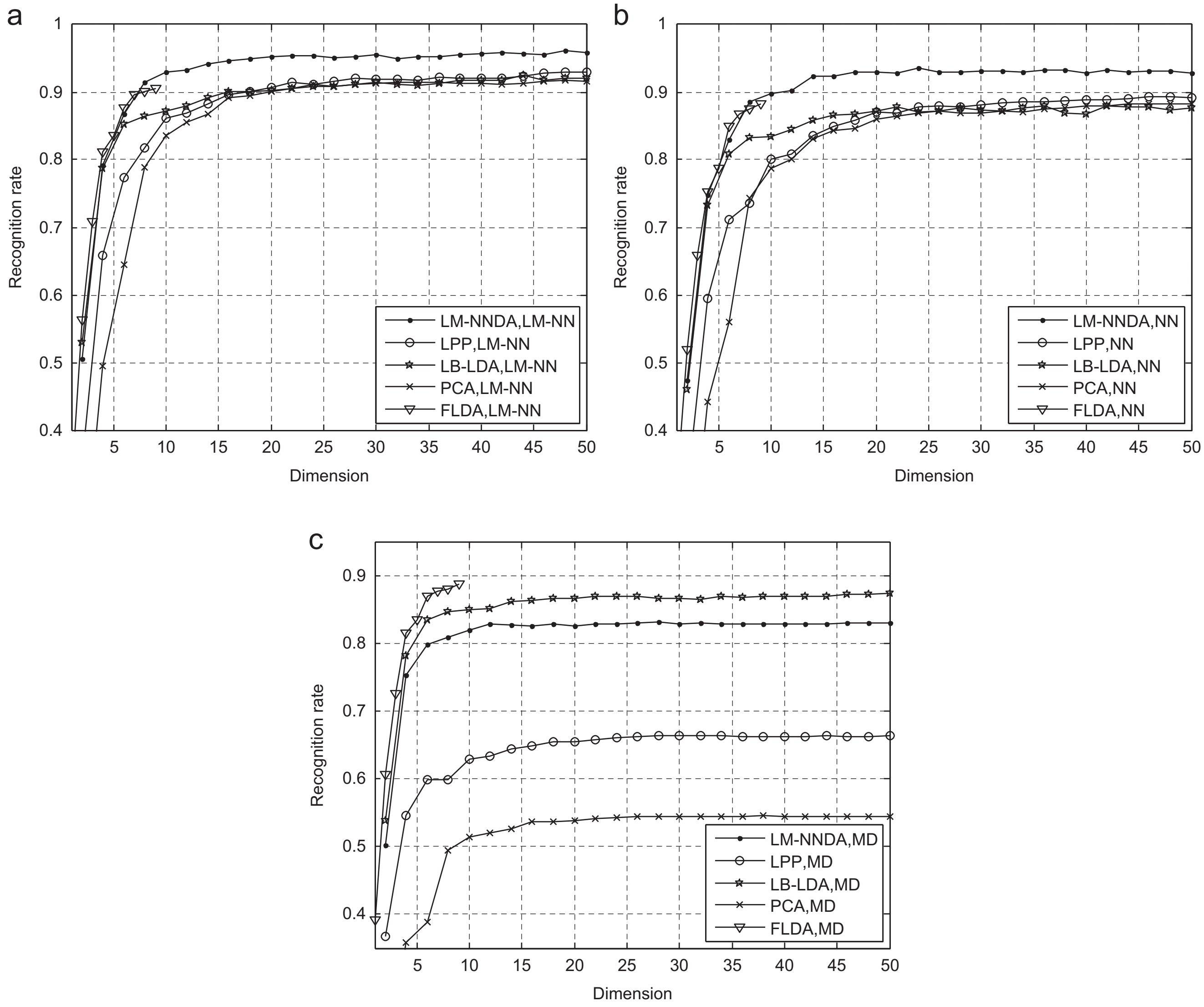

Fig. 6. Recognition rates of PCA, FLDA, LB-LDA, LPP and the proposed LM-NNDA versus dimensions on the test set of the CENPARMI handwritten numeral database: (a) with the LM-NN classifier, (b) with the NN classifier and (c) with the MD classifier.

To verify our analysis in Section 4.3 that LM-NNDA is approaching to FLDA with the increase of the nearest neighbor parameter R_1, we perform an extra test of LM-NNDA in which R_1 is chosen as 199 (note that the number of class training samples is 200). The recognition rates of LM-NNDA with three classifiers are shown in Fig. 7(a). Just as expected, LM-NNDA achieves almost the same recognition results as FLDA when the dimension varies from 1 to 9, no matter what classifier is used. In this case, the first nine LM-NNDA features seem enough for classification, and the remaining features have trivial effect on the recognition performance. To explain this phenomenon, let us provide the values of the criterion in Eq. (13) corresponding to the first 18 projection axes, i.e. the first 18 largest generalized eigenvalues of $\mathbf{S}_b^L\mathbf{X} = \lambda\mathbf{S}_w^L\mathbf{X}$, in Fig. 7(b). It is evident that the eigenvalues except the first nine are very small and invariant. This characteristic of eigenvalues is quite similar to that of FLDA, in which the eigenvalues except the first nine are all zeros.

Now, for further evaluating the performance of the proposed method, we do experiments by 10-fold cross validation. 200 samples are randomly chosen from each class for training, while the remaining 400 samples are used for testing. We run the system 10 times and obtain 10 different training and testing sample sets for performance evaluation. Based on the optimal parameters we obtain on the validation set in the foregoing experiment, we perform PCA, FLDA, LB-LDA, LPP and LM-NNDA with the three classifiers. The average recognition rates and the standard deviations (stds) across ten tests are shown in Table 3.

By comparing the recognition results in the columns of Table 3, we find that for all of the five feature extraction methods, the LM-NN classifier achieves the better results than the other two classifiers. If we compare the performance of the five feature extraction methods based on the LM-NN classifier, LM-NNDA achieves the best result. However, if we do the comparison based on the MD classifier, FLDA achieves the best recognition rate. These results show that LM-NNDA is the most suitable discriminant analysis method for the LM-NN classifier, while FLDA is the most suitable method for the MD classifier.

Here, it should be pointed out that FLDA is the most suitable discriminant analysis method for the MD classifier in theory, but, it is not true vice versa. That is, we cannot say that the MD classifier is most suitable classifier for FLDA. In this experiment, it can be seen that FLDA with the LM-NN classifier achieves better results than with the MD classifier.

Finally let us evaluate the experimental results in Table 3 using a paired *t*-test. If the resulting *p*-value is below the desired significance level (e.g. 0.05), the performance difference between two algorithms is considered to be statistically significant. By this test, we find that under the same LM-NN classifier, the performance of the first-ranked LM-NNDA method is statistically significantly better than that of the second-ranked LB-LDA method at a significance level $p = 5.53 \times 10^{-9}$.

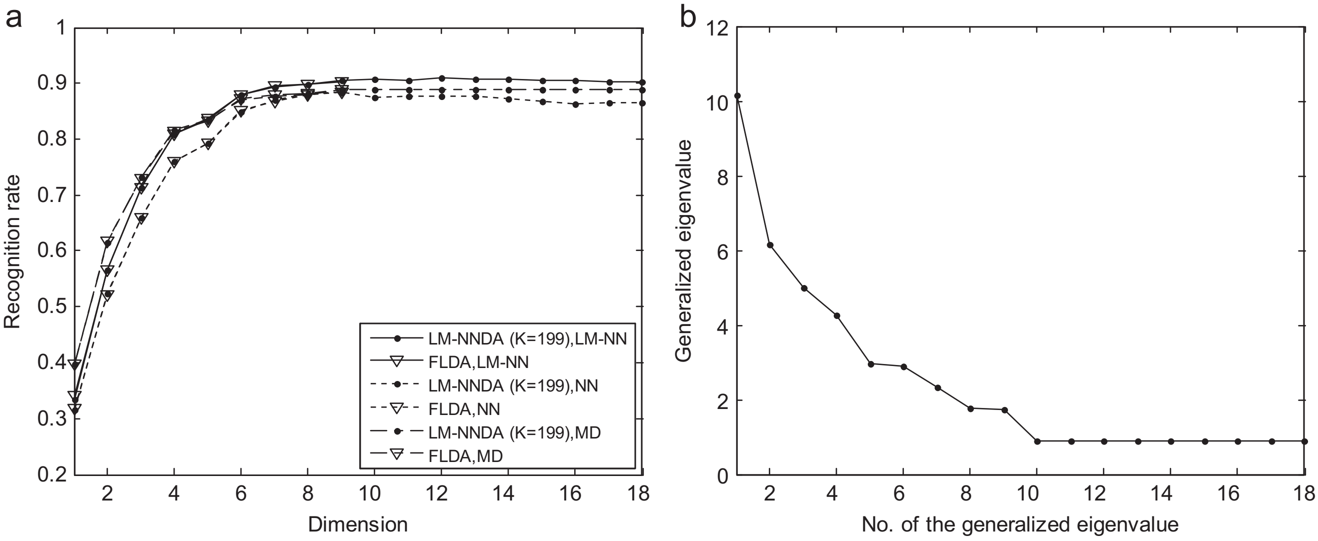

Fig. 7. The performance of LM-NNDA approximates to that of FLDA when R_1 approximates to the training sample number of each class: (a) The recognition rates of LM-NNDA when $R_1 = 199$ (the training sample number of each class is 200) and FLDA and (b) the generalized eigenvalues (i.e. the values of discriminant criterion) corresponding to LM-NNDA.

Table 3
The average recognition rates (%) and the standard deviations (stds) of PCA, FLDA, LB-LDA, LPP and LM-NNDA with each of the three different classifiers using 10-run test on the CENPARMI handwritten numeral database and the corresponding dimensions.

Feature extractor	FLDA	PCA	LB-LDA	LPP	LM-NNDA
Classifier	LM-NN	LM-NN	LM-NN	LM-NN	LM-NN
Recognition rate	89.9 ± 0.45	91.7 ± 0.55	91.8 ± 0.64	91.5 ± 1.04	96.0 ± 0.21
Dimension	9	40	44	50	48
Feature extractor	FLDA	PCA	LB-LDA	LPP	LM-NNDA
Classifier	NN	NN	NN	NN	NN
Recognition rate	87.8 ± 0.37	88.6 ± 0.47	88.2 ± 0.69	88.6 ± 1.12	94.2 ± 0.52
Dimension	9	44	42	48	24
Feature extractor	FLDA	PCA	LB-LDA	LPP	LM-NNDA
Classifier	MD	MD	MD	MD	MD
Recognition rate	87.8 ± 0.36	54.1 ± 1.65	86.6 ± 0.60	67.6 ± 7.06	85.3 ± 0.43
Dimension	9	38	50	34	46

5.2. Experiment using the NUST603 handwritten Chinese character database

The experiment is performed on the NUST603 handwritten Chinese character database which was built in Nanjing University of Science and Technology. The database contains 19 groups of Chinese characters that are collected from bank checks. There are 400 samples for each character and thus 7200 in total. The original 128-dimensional Peripheral feature vectors [29] were extracted for each sample and used in our experiment. The first 100 samples of each class are used for training, the second 100 samples of each class for validation, and the remaining 200 samples for test.

We first apply the three classifiers, the LM-NN classifier, the *K*-NN classifier and the MD classifier, to the original 128-dimensional Peripheral features. The classification results on the validation set are shown in Fig. 8. It seems that the LM-NN classifier consistently outperforms the *K*-NN classifier. From these results, we choose the optimal parameter $K=5$ for the *K*-NN classifier and $K=6$ for the LM-NN classifier. Based on these parameters, we obtain the recognition results of both classifiers on the test set, as listed in Table 4. Table 4 shows that the LM-NN classifier performs much better than the *K*-NN and MD classifiers.

We then apply PCA, FLDA, LB-LDA, LPP and LM-NNDA, respectively, for feature extraction based on the original Peripheral feature vectors. Similar to the experimental methodology adopted in Section 5.1, we use three classifiers: the LM-NN classifier, the NN classifier and the MD classifier for each of the four feature extraction method mentioned. We use the validation set to tune the nearest neighbor parameters for each method as done in the above section and then evaluate their performance on the test set. For each classifier, the recognition rate curve of each method versus the variation of dimensions is shown in Fig. 9.

From Fig. 9, we can draw the follow conclusions. First, LM-NNDA with the LM-NN and NN classifiers achieves better results than PCA, FLDA, LB-LDA and LPP with the same classifiers, which indicates that LM-NNDA, as a feature extractor, is more suitable for the LM-NN (or NN) classifier than the other four methods. Second, FLDA achieves better results than PCA, LPP and LM-NNDA with the MD classifier, which indicates that FLDA is more suitable for the MD classifier than others. Third, the LM-NN classifier is more powerful than the NN classifier and the MD classifier for each feature extraction method. These conclusions are consistent with those we draw on the CENPARMI handwritten numeral database.

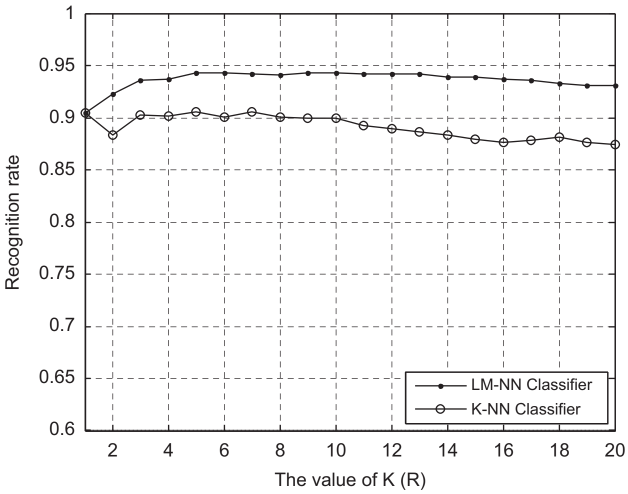

Fig. 8. The recognition rates of the LM-NN classifier and the *K*-NN classifier over the variation of parameters on the NUST603 handwritten Chinese character database.

Table 4
The maximal recognition rates (%) of the three different classifiers based on the original 128-dimensional peripheral feature vectors.

MD	*K*-NN ($K=5$)	LM-NN ($R=6$)
84.3	91.2	95.2

Let us further evaluate the performance of the proposed method using 10-fold cross validation. 100 samples are randomly chosen from each class for training, while the remaining 300 samples are used for testing. We run the system 10 times and obtain 10 different training and testing sample sets for performance evaluation. Based on the optimal parameters we obtain on the validation set in the foregoing experiment, we perform PCA, FLDA, LB-LDA, LPP and LM-NNDA with the three classifiers. The average recognition rates and the standard deviations (stds) across ten tests are shown in Table 5. These results demonstrate again that LM-NNDA is the most suitable discriminant analysis method for the LM-NN classifier, while FLDA is the most suitable method for the MD classifier. In addition, the result of the paired t-test shows that under the LM-NN classifier, the primary LM-NNDA method is statistically significantly better than the secondary LB-LDA method at a significance level $p=3.77\times10^{-5}$.

5.3. Experiment using the ETH80 object category database

The ETH80 database includes 8 categories of biological and artificial objects as shown in Fig. 10. For each category, there are 10 objects that span large in-class variations while still clearly belonging to the category. Each object is represented by 41 images from viewpoints spaced equally over the upper viewing hemisphere (at distances of 22.5–26°). More details about the database can be found in [30,31]. The database is made available in a number of versions for different applications. Here we use the ”cropped-close” version, in which all images are well cropped so that they contain only the object without any border area. In addition, each image is rescaled to a size of 128×128 pixels. In our experiment, all images are converted to grayscale images by averaging the three R, G, and B color components. The size of images is reduced to be 64×64 pixels for computational efficiency. For each object category, we use the first three objects for training, the second three objects for validation and the remaining four objects for testing. Thus, the number of training samples is $8\times3\times41=984$, and the validating samples and testing samples are both $8\times4\times41=1312$.

PCA, FLDA, LB-LDA, LPP and the proposed LM-NNDA are, respectively, used for feature extraction. Since the dimension of the image vector space is much larger than the number of training samples, FLDA, LB-LDA, LPP and LM-NNDA all encounter the ill-posed problem. To address this problem, we first use PCA for dimensionality reduction and then perform these feature extraction methods in the 200-dimensional PCA-transformed space. We use three classifiers: the LM-NN classifier, the NN classifier and the MD classifier for each feature extraction method. We use the validation set to tune the nearest neighbor parameters for each method as done in the above section and then evaluate their performance on the test set. The recognition rate curve of each method versus the variation of dimensions is shown in Fig. 11.

Fig. 11 shows that LM-NNDA with the LM-NN and NN classifiers outperforms PCA, FLDA, LB-LDA and LPP with the same classifiers. This result indicates that LM-NNDA is most suitable feature extractor for the LM-NN (or NN) classifier among the four

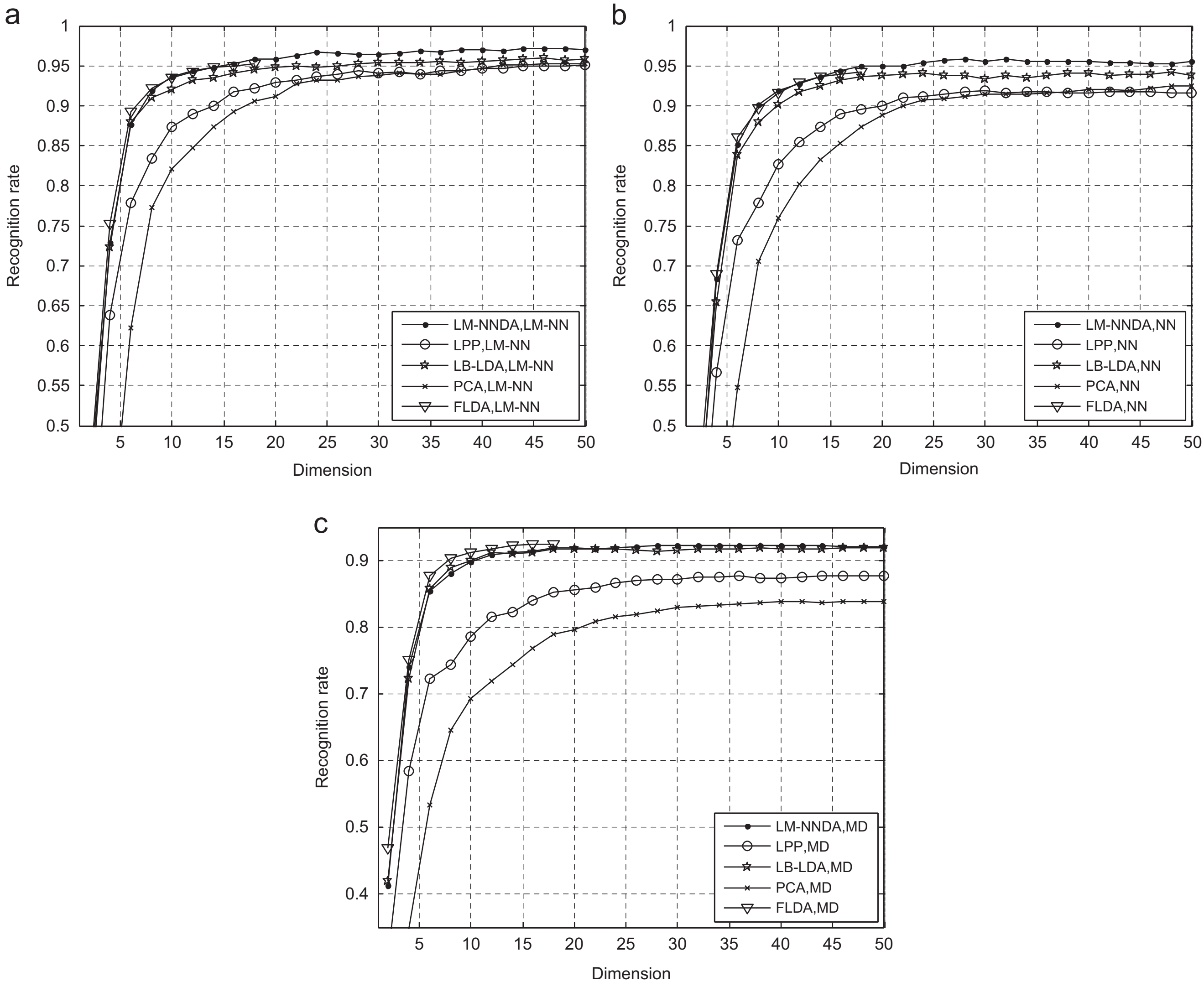

Fig. 9. Recognition rates of PCA, FLDA, LB-LDA, LPP and the proposed LM-NNDA versus dimensions on the NUST603 handwritten Chinese character database: (a) with the LM-NN classifier, (b) with the NN classifier and (c) with the MD classifier.

Table 5
The maximal recognition rates (%) of PCA, FLDA, LB-LDA, LPP and the proposed LM-NNDA with each of the three different classifiers on the NUST603 handwritten Chinese character database and the corresponding dimensions.

Feature extractor	FLDA	PCA	LB-LDA	LPP	LM-NNDA
Classifier	LM-NN	LM-NN	LM-NN	LM-NN	LM-NN
Recognition rate	95.07 ± 0.32	94.57 ± 0.30	95.50 ± 0.30	94.63 ± 0.36	*96.37 ± 0.33*
Dimension	18	46	46	50	48
Feature extractor	FLDA	PCA	LB-LDA	LPP	LM-NNDA
Classifier	NN	NN	NN	NN	NN
Recognition rate	93.75 ± 0.37	91.55 ± 0.39	93.01 ± 0.17	90.37 ± 0.71	*94.85 ± 0.21*
Dimension	18	48	48	30	28
Feature extractor	FLDA	PCA	LB-LDA	LPP	LM-NNDA
Classifier	MD	MD	MD	MD	MD
Recognition rate	*92.41 ± 0.27*	83.36 ± 0.55	92.22 ± 0.38	87.10 ± 0.67	92.40 ± 0.34
Dimension	18	46	38	36	32

methods. In addition, FLDA achieves better results than PCA, LB-LDA, LPP and LM-NNDA with the MD classifier, which verifies that FLDA is the most suitable feature extractor for the MD classifier.

Table 6 shows the average recognition rates and the standard deviations (stds) across 10-fold tests. In each test, three objects are randomly chosen for training, the remaining seven objects for testing. These results demonstrate again that LM-NNDA is the best for the LM-NN classifier, while FLDA is the best for the MD classifier. In addition, the result of the paired *t*-test shows that under the LM-NN classifier, the primary LM-NNDA method is statistically significantly better than the secondary LB-LDA method at a significance level $p = 3.16 \times 10^{-4}$.

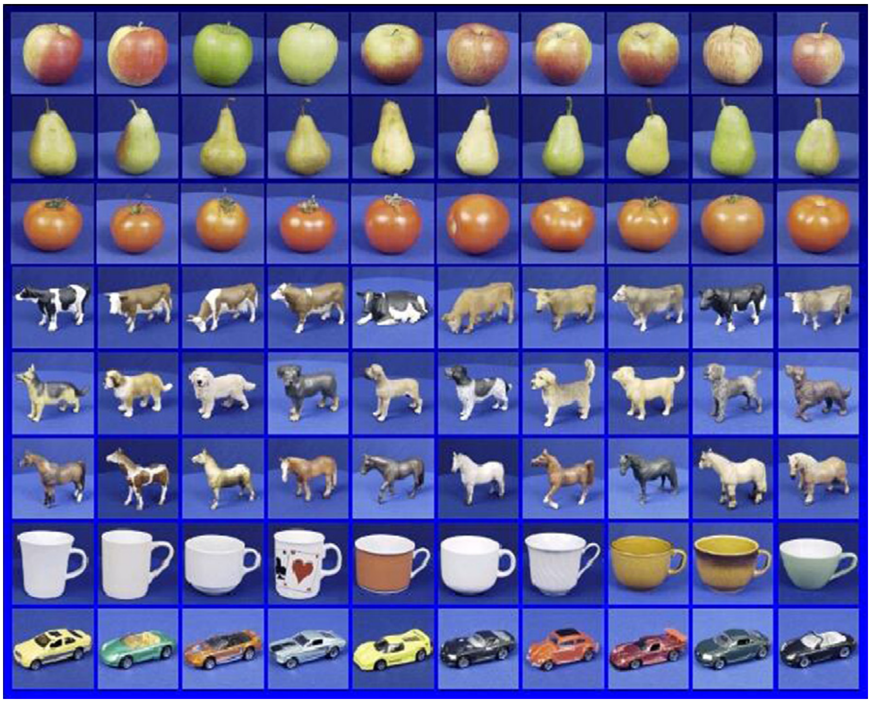

Fig. 10. The example figures of 8 categories of objects in the ETH80 database.

5.4. *Experiment using the FERET database*

The FERET face image database is a popular database for testing and evaluating state-of-the-art face recognition algorithms [32,33]. Our experiment uses a subset of the database, which includes 1000 images of 200 individuals (each one has 5 images). It is composed of the images whose names are marked with two-character strings: "*ba*", "*bj*", "*bk*", "*be*" and "*bf*". This subset involves variations in facial expression, illumination, and pose. In our experiment, the facial portion of each original image was automatically cropped based on the location of eyes and mouth, and the cropped image was resized to 80 × 80 pixels and further pre-processed by histogram equalization. Some sample images of one person are shown in Fig. 12.

In the first experiment, we use the first two images (i.e. "*ba*" and "*bj*") of each class to form the training set, and the remaining three images (i.e. "*bk*", "*be*" and "*bf*") of each class to form the validation set. PCA, FLDA, LB-LDA, LPP and the proposed LM-NNDA are used for feature extraction. Since the dimension of the image vector space is much larger than the number of training samples, FLDA, LB-LDA, LPP and LM-NNDA all encounter the ill-posed problem. To address this problem, we first use PCA for dimensionality reduction and then perform FLDA, LB-LDA, LPP and LM-NNDA in the 120-dimensional PCA-transformed space. Since there are only two training samples per class, the nearest neighbor parameter R_1 is chosen as 1. Thus LM-NNDA is actually NNDA in this case. We use the NN classifier for each feature extraction method. The recognition rate curve of each method versus the variation of dimensions is shown in Fig. 13. The maximal recognition rate of each method and the corresponding dimension are shown in Table 7. From Fig. 13, we can see that NNDA consistently outperforms the other four methods on the validation set, irrespective of the variation in the dimensions.

We further perform experiments by 10-run tests. In each run, we randomly choose two images from each class for training, and the remaining images for test. Based on the optimal dimensions we obtain on the validation set in Table 7, for each method, we perform PCA, FLDA, LB-LDA, LPP and LM-NNDA with the three classifiers. The average recognition rates and the standard deviations (stds) across ten tests are shown in Table 8. These results indicate that NNDA, a nearest neighbor rule induced discriminant analysis, is the most suitable method for the NN classifier. In addition, from the paired *t*-test, we know that NNDA significantly outperforms the secondary FLDA method at a significance level $p = 1.61 \times 10^{-5}$.

6. Conclusion, discussion and future work

This paper introduces a new concept of designing a discriminant analysis method, which starts from a local mean based nearest neighbor (LM-NN) classifier and uses its decision rule to direct the design of a discriminant analysis method. The derived discriminant analysis method, LM-NNDA, is the most suitable feature extractor for the LM-NN classifier in theory. This has been demonstrated by our experimental results on four databases: the CENPARMI handwritten numeral database, the NUST603 handwritten Chinese character database, the ETH80 object category database and the FERET face image database. In addition, we show that the classical Fisher linear discriminant analysis (FLDA) is a minimum distance classifier (or called Nearest Class-mean classifier) induced discriminant analysis method. This judgment was also verified by our experimental results.

The LM-NNDA based feature extractor closely connects to the NN classifier. The NN classifier has an asymptotical average error rate connected to Bayes error via lower and upper bounds as shown in Eq. (2). It can be expected that the projection matrix of LM-NNDA is Bayes suboptimal, that is, the asymptotical average error rate based on the LM-NNDA generated features has a lower and upper bound of the Bayes error. Recently Petridis and Perantonis gave the concept of the Bayes optimal projection matrix and provided a theoretical framework to analyze the Bayes optimality of linear discriminant analysis methods [34]. Hamsici and Martinez present an algorithm that can provide the one-dimensional subspace where the Bayes error is minimized for the C class problem with homoscedastic Gaussian distributions and further extend the algorithm to suit for more general case

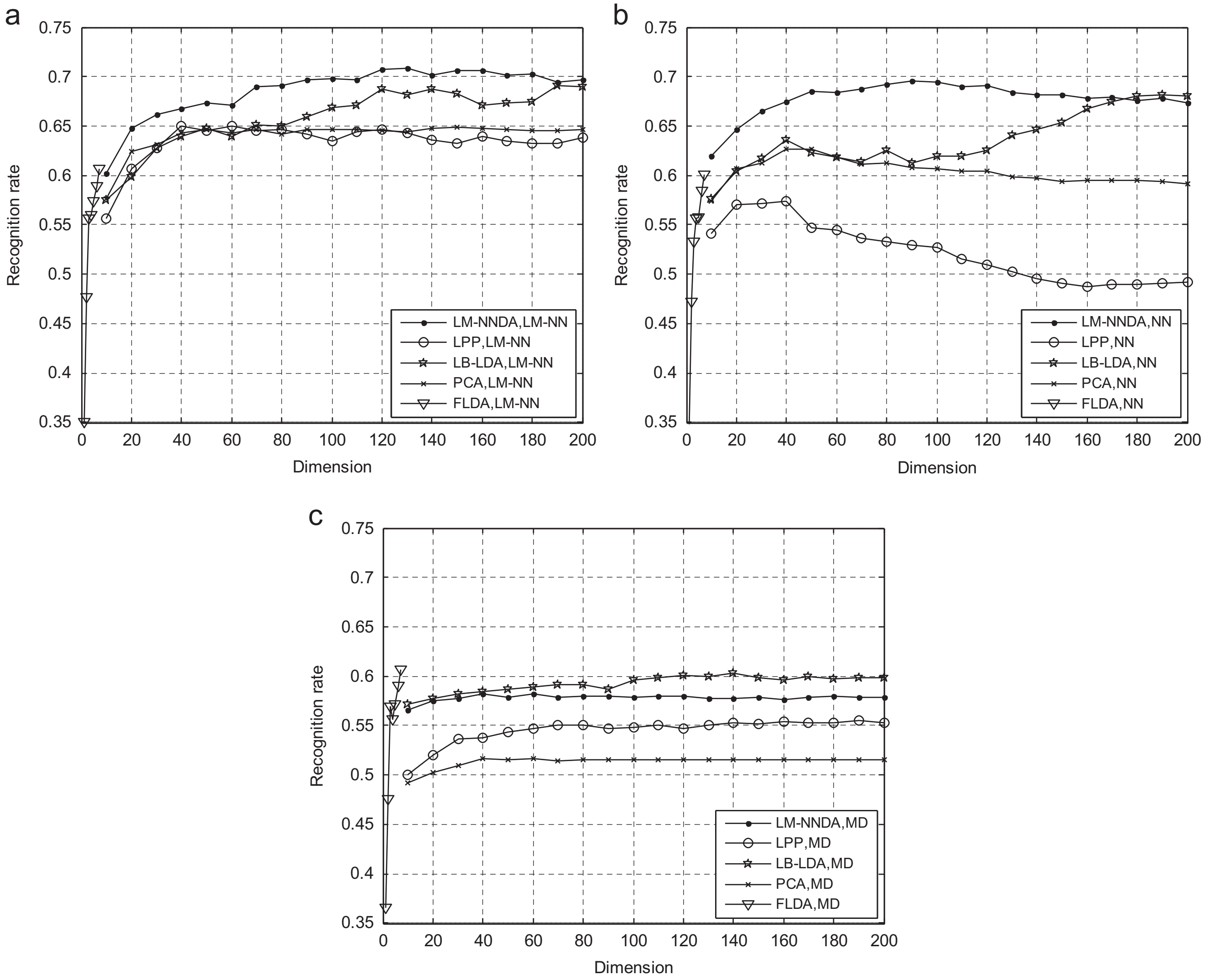

Fig. 11. Recognition rates of PCA, FLDA, LB-LDA, LPP and the proposed LM-NNDA versus dimensions on the ETH80 object category database: (a) with the LM-NN classifier, (b) with the NN classifier and (c) with the MD classifier.

Table 6
The maximal recognition rates (%) of PCA, FLDA, LB-LDA, LPP and the proposed LM-NNDA with each of the three different classifiers on the ETH80 object category database and the corresponding dimensions.

Feature extractor	FLDA	PCA	LB-LDA	LPP	LM-NNDA
Classifier	LM-NN	LM-NN	LM-NN	LM-NN	LM-NN
Recognition rate	62.49 ± 2.95	68.85 ± 3.45	71.86 ± 3.87	66.97 ± 2.61	*75.19 ± 2.79*
Dimension	7	150	190	40	130
Feature extractor	FLDA	PCA	LB-LDA	LPP	LM-NNDA
Classifier	NN	NN	NN	NN	NN
Recognition rate	60.79 ± 3.32	66.10 ± 3.21	69.65 ± 2.79	61.89 ± 3.66	*74.02 ± 3.21*
Dimension	7	40	190	40	90
Feature extractor	FLDA	PCA	LB-LDA	LPP	LM-NNDA
Classifier	MD	MD	MD	MD	MD
Recognition rate	*62.20 ± 2.48*	50.07 ± 3.96	61.97 ± 2.68	56.68 ± 2.73	59.63 ± 2.97
Dimension	7	40	140	190	40

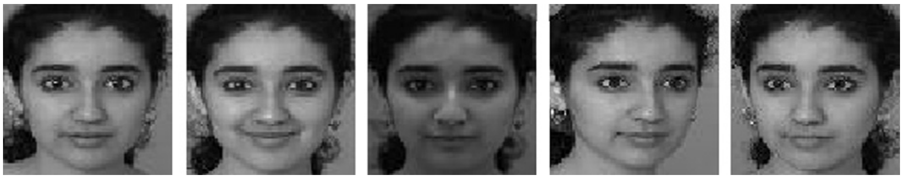

Fig. 12. Samples of the cropped images in the FERET database.

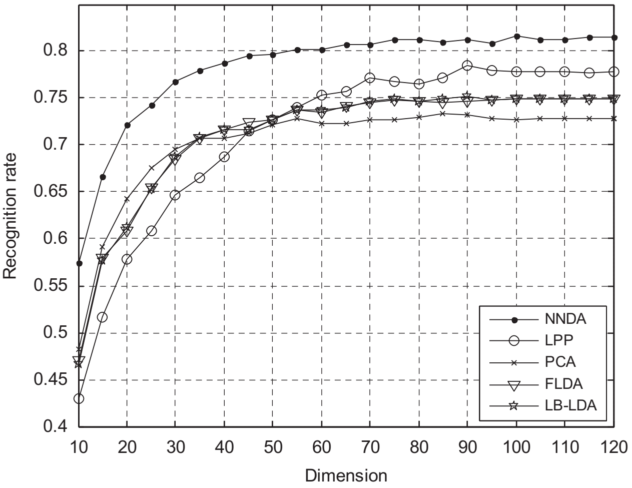

Fig. 13. Recognition rates of PCA, FLDA, LB-LDA, LLP and the proposed NNDA method with the NN classifier versus dimensions on the validation set of the FERET database.

Table 7
The maximal recognition rates (%) of PCA, FLDA, LB-LDA, LLP and the proposed NNDA method with the NN classifier on the validation set of the FERET database and the corresponding dimensions.

Feature extractor	FLDA	PCA	LB-LDA	LPP	NNDA
Recognition rate	75.0	73.3	75.2	78.5	81.7
Dimension	100	85	90	90	100

Table 8
The average recognition rates (%) and the standard deviations (stds) of PCA, FLDA, LB-LDA, LLP and the proposed NNDA method with the NN classifier using 10-run tests on the FERET database.

FLDA	PCA	LB-LDA	LPP	NNDA
76.06 ± 1.88	72.40 ± 2.38	76.02 ± 1.79	72.48 ± 5.13	82.86 ± 1.27

with heteroscedastic distributions and to obtain the d-dimensional solutions [35]. Generally, discussing the Bayes optimality of a linear feature extraction method needs some assumptions on the probability distribution of data. For example, we have shown that LM-NNDA is Bayes optimal under the assumption that all set of R local samples share a normal distribution with an identical covariance matrix. However, how to analyze the Bayes optimality of the proposed method without any assumption is still open; some new ideas and theoretical tools are required and we expect them to be developed in the future.

Acknowledgments

The authors would like to thank the anonymous reviewers for their critical and constructive comments and suggestions. This work was partially supported by the Program for New Century Excellent Talents in University of China, the NUST Outstanding Scholar Supporting Program, the National Science Foundation of China under Grant nos. 60973098 and 60632050, and the Hong Kong RGC General Research Fund (PolyU 5351/08E).

References

[1] T.M. Cover, P.E. Hart, Nearest neighbor pattern classification, IEEE Transactions on Information Theory 13 (1) (1967) 21–27.

[2] K. Fukunaga, Introduction to Statistical Pattern Recognition, second edition, Academic Press, 1990.

[3] S.Z. Li, Juwei Lu, Face recognition using the nearest feature line method, IEEE Transactions on Neural Networks 10 (2) (1999) 439–443.

[4] J.-T. Chien, C.-C. Wu, Discriminant waveletfaces and nearest feature classifiers for face recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (12) (2002) 1644–1649.

[5] W. Zheng, L. Zhao, C. Zou, Locally nearest neighbour classifiers for pattern classification, Pattern Recognition 37 (6) (2004) 1307–1309.

[6] Z. Lou, Z. Jin, Novel adaptive nearest neighbor classifiers based on hit-distance, in: Proceedings of the 18th International Conference on Pattern Recognition (ICPR 2006), vol. 3, August 20–24, 2006, pp. 87–90.

[7] H. Du, Y.Q. Chen, Rectified nearest feature line segment for pattern classification, Pattern Recognition 40 (2007) 1486–1497.

[8] Y. Mitani, Y. Hamamoto, A local mean-based nonparametric classifier, Pattern Recognition Letters 27 (10) (2006) 1151–1159.

[9] Daniel L. Swets, John Weng, Using discriminant eigenfeatures for image retrieval, IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (8) (1996) 831–836.

[10] P.N. Belhumeur, J.P. Hespanha, D.J. Kriengman, Eigenfaces vs. Fisherfaces: recognition using class specific linear projection, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (7) (1997) 711–720.

[11] C.J. Liu, H. Wechsler, Robust coding schemes for indexing and retrieval from large face databases, IEEE Transactions on Image Processing 9 (1) (2000) 132–137.

[12] H. Yu, J. Yang, A direct LDA algorithm for high-dimensional data—with application to face recognition, Pattern Recognition 34 (10) (2001) 2067–2070.

[13] J. Yang, J.Y. Yang, Why can LDA be performed in PCA transformed space?, Pattern Recognition 36 (2) (2003) 563–566.

[14] W. Zhao, R. Chellappa, J. Phillips, Subspace linear discriminant analysis for face recognition, Technical Report, CS-TR4009, University of Maryland, 1999.

[15] J. Ye, R. Janardan, C. Park, H. Park, An optimization criterion for generalized discriminant analysis on undersampled problems, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (8) (2004) 982–994.

[16] H. Cevikalp, M. Neamtu, M. Wilkes, A. Barkana, Discriminative common vectors for face recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (1) (2005) 4–13.

[17] S. Mika, G. Rätsch, J. Weston, B. Schölkopf, K.-R. Müller Fisher discriminant analysis with kernels, in: IEEE International Workshop on Neural Networks for Signal Processing, vol. IX, Madison, USA, August 1999, pp. 41–48.

[18] G. Baudat, F. Anouar, Generalized discriminant analysis using a kernel approach, Neural Computation 12 (10) (2000) 2385–2404.

[19] J. Lu, K.N. Plataniotis, A.N. Venetsanopoulos, Face recognition using kernel direct discriminant analysis algorithms, IEEE Transactions on Neural Networks 14 (1) (2003) 117–126.

[20] J. Yang, A.F. Frangi, J.-Y. Yang, D. Zhang, Z. Jin, KPCA plus LDA: a complete kernel Fisher discriminant framework for feature extraction and recognition, IEEE Transactions on Pattern Analysis Machine Intelligence 27 (2) (2005) 230–244.

[21] T.-K. Kim, J. Kittler, Locally linear discriminant analysis for multimodally distributed classes for face recognition with a single model image, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (3) (2005) 318–327.

[22] X. He, S. Yan, Y. Hu, P. Niyogi, H.-J. Zhang, Face recognition using Laplacian faces, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (3) (2005) 328–340.

[23] H.-T. Chen, H.-W. Chang, T.-L. Liu, Local discriminant embedding and its variants, in: IEEE Conference on Computer Vision and Pattern Recognition 2005 (CVPR 2005), pp. 846–853.

[24] S. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang, S. Lin, Graph embedding and extension: a general framework for dimensionality reduction, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (1) (2007) 40–51.

[25] P. Vincent, Y. Bengio, *K*-local hyperplane and convex distance nearest neighbor algorithms, in: Advances in Neural Information Processing Systems (NIPS2002), 2002.

[26] G.H. Golub, C.F. Van Loan, Matrix Computations, Third edition, The Johns Hopkins University Press, Baltimore, London, 1996.

[27] X.P. Qiu, L.D. Wu, Face recognition by stepwise nonparametric margin maximum criterion, in: Proceedings of the IEEE Conference on Computer Vision, Beijing, China, October 2005.

[28] S.X. Liao, M. Pawlak, On image analysis by moments, IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (3) (1996) 254–266.

[29] Y.H. Tseng, C.C. Kuo, H.J. Lee, Speeding up Chinese character recognition in an automatic document reading system, Pattern Recognition 31 (11) (1998) 1601–1612.

[30] B. Leibe, B. Schiele, Analyzing appearance and contour based methods for object categorization, in: International Conference on Computer Vision and Pattern Recognition (CVPR'03), Madison, Wisconsin, June 2003.

[31] B. Leibe, The ETH-80 Image Set. Available from: 〈http://www.mis.informatik.tu-darmstadt.de/Research/Projects/categorization/eth80-db.html〉.

[32] P.J. Phillips, H. Moon, S.A. Rizvi, P.J. Rauss, The FERET evaluation methodology for face-recognition algorithms, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (10) (2000) 1090–1104.

[33] P.J. Phillips, H. Wechsler, J. Huang, P. Rauss, The FERET database and evaluation procedure for face recognition algorithms, Image and Vision Computing 16 (5) (1998) 295–306.

[34] S. Petridis, S.T. Perantonis, On the relation between discriminant analysis and mutual information for supervised linear feature extraction, Pattern Recognition 37 (5) (2004) 857–874.

[35] Onur C. Hamsici, Aleix M. Martinez, Bayes optimality in linear discriminant analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (4) (2008) 647–657.

[36] R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, 2nd Edition, John Wiley & Sons, Inc., 2000.

[37] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (2000) 2323–2326.

[38] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Computation 15 (6) (2003) 1373–1396.

[41] H.S. Seung, D.D. Lee, The manifold ways of perception, Science 290 (2000) 2268–2269.

[42] J.B. Tenenbaum, V. de Silva, J.C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (2000) 2319–2323.

[43] Nearest neighbor search. Available from: 〈http://en.wikipedia.org/wiki/Nearest_neighbor_search〉.

[46] T. Hastie, R. Tibshirani, Discriminant Adaptive Nearest Neighbor Classification, IEEE Transactions on Pattern Analysis on Machine Intelligence 18 (6) (1996) 607–615.

[47] C. Domeniconi, J. Peng, D. Gunopulos, Locally adaptive metric nearest-neighbor classification, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (9) (2002) 1281–1285.

[48] M. Bressan, J. Vitria, Nonparametric discriminant analysis and nearest neighbor classification, Pattern Recognition Letters 24 (2003) 2743–2749.

[49] H. Zhang, A.C. Berg, M. Maire, J. Malik, SVM-KNN: Discriminative Nearest Neighbor Classification for Visual Category Recognition, CVPR, 2006.

Jian Yang received the B.S. degree in mathematics from the Xuzhou Normal University in 1995. He received the MS degree in applied mathematics from the Changsha Railway University in 1998 and the Ph.D. degree from the Nanjing University of Science and Technology (NUST), on the subject of pattern recognition and intelligence systems in 2002. In 2003, he was a postdoctoral researcher at the University of Zaragoza, and in the same year, he was awarded the RyC program Research Fellowship sponsored by the Spanish Ministry of Science and Technology. From 2004 to 2006, he was a Postdoctoral Fellow at Biometrics Centre of Hong Kong Polytechnic University. From 2006 to 2007, he was a Postdoctoral Fellow at Department of Computer Science of New Jersey Institute of Technology. Now, he is a professor in the School of Computer Science and Technology of NUST. He is the author of more than 50 scientific papers in pattern recognition and computer vision. His research interests include pattern recognition, computer vision and machine learning. Currently he is an associate editor of Pattern Recognition Letters and Neurocomputing, respectively.

Lei Zhang received the B.S. degree in 1995 from Shenyang Institute of Aeronautical Engineering, Shenyang, P.R. China, the M.S. and Ph.D degrees in Automatic Control Theory and Engineering from Northwestern Polytechnical University, Xi'an, P.R. China, respectively, in 1998 and 2001. From 2001 to 2002, he was a research associate in the Dept. of Computing, The Hong Kong Polytechnic University. From January 2003 to January 2006 he worked as a Postdoctoral Fellow in the Department of Electrical and Computer Engineering, McMaster University, Canada. Since Jan. 2006, he has been an Assistant Professor in the Dept. of Computing, The Hong Kong Polytechnic University. His research interests include Image and Video Processing, Biometrics, Pattern Recognition, Multisensor Data Fusion and Optimal Estimation Theory, etc.

Jing-yu Yang received the B.S. Degree in Computer Science from Nanjing University of Science and Technology (NUST), Nanjing, China. From 1982 to 1984 he was a visiting scientist at the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign. From 1993 to 1994 he was a visiting professor at the Department of Computer Science, Missuria University. And in 1998, he acted as a visiting professor at Concordia University in Canada. He is currently a professor and Chairman in the department of Computer Science at NUST. He is the author of over 300 scientific papers in computer vision, pattern recognition, and artificial intelligence. He has won more than 20 provincial awards and national awards. His current research interests are in the areas of pattern recognition, robot vision, image processing, data fusion, and artificial intelligence.

David Zhang graduated in Computer Science from Peking University. He received his MSc in Computer Science in 1982 and his Ph.D. in 1985 from the Harbin Institute of Technology (HIT). From 1986 to 1988 he was a Postdoctoral Fellow at Tsinghua University and then an Associate Professor at the Academia Sinica, Beijing. In 1994 he received his second PhD in Electrical and Computer Engineering from the University of Waterloo, Ontario, Canada. Currently he is a Chair Professor at the Hong Kong Polytechnic University where he is the Founding Director of the Biometrics Technology Centre (UGC/CRC) supported by the Hong Kong SAR Government. He also serves as Adjunct Professor in Tsinghua University, Shanghai Jiao Tong University, Beihang University, Harbin Institute of Technology, and the University of Waterloo. He is the Founder and Editor-in-Chief, International Journal of Image and Graphics (IJIG); Book Editor, Springer International Series on Biometrics (KISB); Organizer, the International Conference on Biometrics Authentication (ICBA); Associate Editor of several international journals including IEEE Trans on SMC-A/SMC-C; Technical Committee Chair of IEEE CIS and the author of more than 10 books and 160 journal papers. Professor Zhang is a Croucher Senior Research Fellow, Distinguished Speaker of the IEEE Computer Society, and a Fellow of the International Association of Pattern Recognition (IAPR), and IEEE Fellow.