2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Hidden Space Discriminant Neighborhood Embedding

Chuntao Ding
School of Computer Science and Technology & Provincial Key laboratory for Computer Information Processing, Soochow University, Suzhou 215006, Jiangsu, China
Email: 20124227036@suda.edu.cn

Li Zhang
School of Computer Science and Technology & Provincial Key laboratory for Computer Information Processing, Soochow University, Suzhou 215006, Jiangsu, China
Email: zhangliml@suda.edu.cn

Bangjun Wang
School of Computer Science and Technology & Provincial Key laboratory for Computer Information Processing, Soochow University, Suzhou 215006, Jiangsu, China
Email: wangbangjun@suda.edu.cn

***Abstract*—Discriminant neighborhood embedding (DNE) algorithm is one of supervised linear dimensionality reduction methods. Its nonlinear version kernel discriminant neighborhood embedding (KDNE) is expected to behave well on classification tasks. However, since KDNE constructs an adjacent graph in the original space, the adjacency graph could not represent the adjacent information in the kernel mapping space. By introducing hidden space, this paper proposes a novel nonlinear method for DNE, called hidden space discriminant neighborhood embedding (HDNE). This algorithm first maps the data in the original space into a high dimensional hidden space by a set of nonlinear hidden functions, and then builds an adjacent graph incorporating neighborhood information of the dataset in the hidden space. Finally, DNE is used to find a transformation matrix which would map the data in the hidden space to a low-dimensional subspace. The proposed method is applied to ORL face and MNIST handwritten digit databases. Experimental results show that the proposed method is efficiency for classification tasks.**

I. Introduction

Dimensionality reduction methods have been attracted a lot of attention in machine learning, pattern recognition and computer vision etc. As one of important preprocessing steps in the analysis of high dimensional data, dimensionality reduction usually makes the data in a high dimensional space embed in a relatively low dimensional space, meanwhile, with most of the original data information preserved [1] [2]. Usually, dimensionality reduction methods can be divided into two groups, or linear and non-linear ones [4][5][6][7][8][9][10][11][12][13][14][15][16].

The most classic linear dimensionality reduction method is principal components analysis (PCA), of which the variance of data is used to measure useful information [20]. Typically, the larger the variance of data in some direction is, the more information this direction has; otherwise, the less information and value it has.

Since S. Roweis et al. proposed locally linear embedding (LLE) algorithm [3], manifold learning representing nonlinear dimensionality reduction methods quickly attracted attention of so many researchers. For the advantage of both linear dimensionality reduction and manifold learning, locality preserving projection (LPP), regarded as an upgrade version of PCA, was proposed [4]. As an unsupervised dimensionality reduction method, it could maximally keep the neighborhood structure of a high dimensional dataset. If points are close to each other in both the original space, they remain a relatively close distance after reducing dimensionality so as to preserve the local structure. Therefore, LPP is able to find a better projection direction for the data belonging to different classes with a far distance between each other. Some manifold learning methods, such as LLE, cannot yield a projection matrix, so they cannot perform incremental learning for new data. To cover this shortage, neighborhood preserving embedding (NPE) was proposed, which is a linear approximation of LLE and able to learn a projection matrix [5].

Usually, classification is a supervised learning with prior knowledge of class information. However, manifold dimensionality reduction methods discussed above are all unsupervised so that they cannot make full use of the prior knowledge. To remedy this, Zhang et al. proposed a supervised linear dimensionality reduction method, called discriminant neighborhood embedding (DNE) [16]. In DNE, if the points belonging to the same class are close to each other in original space, they would still remain a relatively close distance after reducing dimensionality. While if the points belonging to the different classes are close to each other in original space, they would remain a relatively far distance after reducing dimensionality. By introducing kernel tricks into DNE, a non-linear version called kernel DNE (KDNE) was proposed [25], where kernel function must satisfy Mercer's condition [17] and [18]. Nevertheless, being similar to DNE, KDNE only constructs the adjacent graph of original space without taking into account one of mapping space so that local geometric structure cannot be preserved efficiently when learning dimensionality reduction of the transition matrix.

Considering that DNE cannot get a better projection with linearly non-separable samples and KDNE cannot employ neighborhood relationships efficiently in a high dimensional space, we introduce the conception of hidden space. By using a nonlinear hidden function, the data in the original space are mapped into a high dimensional space. As a consequence, some linearly non-separable samples in a low dimensional space are now separable [19]. The novel method

978-1-4799-1484-5/14/$31.00 ©2014 IEEE

is called hidden space discriminant neighborhood embedding (HDNE), which is also a nonlinear extension of DNE.

HDNE first maps the data in the original space into the the high dimensional hidden space in which the data would be linearly separable, and then builds its adjacent graph so that the local relationships for samples can be preserved in the hidden space. Reducing the dimensionality of samples in the hidden space by applying DNE can make the samples be linearly separable not only in the hidden space but also in the discriminant subspace. As a result, the recognition rate could be significantly improved. Experimental results on artificial and real-world datasets show that HDNE has higher recognition rates.

The remainder of the paper is organized as follows. In Section 2, we briefly review the DNE and KDNE. Section 3 presents HDNE. Simulation experiments are given in Section 4 and conclusions are provided in Section 5.

II. Related Works

In this section, DNE method and KDNE method will be reviewed briefly.

A. Discriminant Neighborhood Embedding

To exploit the class information efficiently, Zhang et al. proposed DNE which requires to build an adjacent graph between the samples in an original space and meanwhile tries to preserve the adjacent relationships in a low dimensional space. If the points belonging to the same class are close to each other in the high dimensional space, they would remain a relatively close distance in a discriminant subspace. If the points belonging to the different classes are close to each other in the high dimensional space, they would be separated in a discriminant subspace. Next, we will give a brief introduction of DNE algorithm.

Given a set of training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$, where $\mathbf{x}_i \in R^m$, $y_i \in \{1, 2, ..., c\}$ is the label of $\mathbf{x}_i$, and c, N and m denote the number of classes, the number of samples and dimensionality, respectively. The goal of DNE is to find a projection matrix $\mathbf{A}$. If any two samples $\mathbf{x}_i$ and $\mathbf{x}_j$ belonging to the same class are close, $\mathbf{v}_i = \mathbf{A}^T\mathbf{x}_i$ and $\mathbf{v}_j = \mathbf{A}^T\mathbf{x}_j$ are close, too. Of course, if they belong to different classes, the distance between them would become far after projection. The projection matrix is represented as $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_d] \in R^{m\times d}$, where $d < m$ and the vectors $\mathbf{a}_i \in R^m$ are independent of each other. The detail procedure of DNE is listed in Algorithm 1.

B. Kernel Discriminant Neighborhood Embedding

Given $\mathbf{x}, \mathbf{z} \in X \subseteq R^m$ and nonlinear function Φ, we can map $\mathbf{x}$ and $\mathbf{z}$ in the input space X into a feature space F, where $F \subseteq R^M$ and $m \ll M$. According to the Mercer theorem, we have

$$k(\mathbf{x}, \mathbf{z}) = \Phi(\mathbf{x})^T\Phi(\mathbf{z})$$

where $k(\mathbf{x}, \mathbf{z})$ denotes a Mercer kernel function which makes M-dimensional inner product operation in a high dimensional space change to be m-dimensional calculus of function in a low-dimensional space.

Algorithm 1 DNE

Input: Training sample matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in R^{m\times N}$, and the dimensionality of discriminant subspace d;
Output: Projection matrix $\mathbf{A}$;

1). Construct the adjacent graph $\mathbf{F}$, which is defined as:

$$F_{ij} = \begin{cases} +1, & \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are neighbors and } y_i = y_j \\ -1, & \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are neighbors and } y_i \neq y_j \\ 0, & \text{otherwise} \end{cases}$$

2). Solve the following optimization problem:

$$\min_{\mathbf{A}} trace(\mathbf{A}^T\mathbf{X}(\mathbf{S}-\mathbf{F})\mathbf{X}^T\mathbf{A})$$
$$\text{s.t.} \quad \mathbf{a}_i{}^T\mathbf{a}_i = 1, \mathbf{a}_i{}^T\mathbf{a}_j = 0, i \neq j, i, j = 1, \ldots, d$$

where $\mathbf{S}$ is a diagonal matrix and its entries are $\mathbf{S}_{ii} = \sum_j \mathbf{F}_{ji}$.
The projection matrix $\mathbf{A}$ can be obtained by computing the eigenvalue problem of $\mathbf{X}(\mathbf{S}-\mathbf{F})\mathbf{X}^T\mathbf{A} = \lambda\mathbf{A}$.
Let eigenvalues be λ_i and their corresponding eigenvectors be $\mathbf{a}_i$. Assume that $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_m$.

3). Return $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_d]$

As a result, the problems such as curse of dimensionality are solved skillfully. Next, three common kernels are presented below [22]. Polynomial kernel is

$$k(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i{}^T\mathbf{x}_j + c)^p \tag{1}$$

where p is parameter of this kernel. Gaussian kernel is

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2/s^2) \tag{2}$$

where $s > 0$ is parameter of this kernel. Linear kernel is

$$k(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i{}^T\mathbf{x}_j \tag{3}$$

By introducing kernel tricks, DNE could be generalized to its nonlinear version, or KDNE. The goal of KDNE is also to find a projection matrix $\mathbf{A} \in R^{N\times d}$ which cannot be obtained explicitly. Fortunately, we could get samples in the discriminant subspace space by using an auxiliary matrix $\mathbf{B} \in R^{N\times d}$. Namely, $\mathbf{v}_j = \mathbf{B}^T\mathbf{K}_{\cdot j}$, where $\mathbf{K}$ is a kernel Gram matrix with $K_{ij} = k\left(\mathbf{x}_i, \mathbf{x}_j\right)$. The detail for KDNE is given in Algorithm 2.

Algorithm 2 KDNE

Input: A training set $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$, and the dimension of discriminant subspace d
Output: Auxiliary matrix $\mathbf{B}$;

1). Construct the adjacent graph $\mathbf{F}$ which is defined as:

$$F_{ij} = \begin{cases} +1, & \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are neighbors and } y_i = y_j \\ -1, & \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are neighbors and } y_i \neq y_j \\ 0, & \text{otherwise} \end{cases}$$

2). Solve the following optimization problem:

$$\min \quad trace(\mathbf{A}^T(\mathbf{S}-\mathbf{F})\mathbf{K}^T\mathbf{A})$$
$$\text{s.t.} \quad \mathbf{a}_i{}^T\mathbf{a}_i = 1, \mathbf{a}_i{}^T\mathbf{a}_j = 0, i \neq j, i, j = 1, \ldots, d$$

where $\mathbf{S}$ is a diagonal matrix and its entries are $\mathbf{S}_{ii} = \sum_j \mathbf{F}_{ji}$.
The auxiliary matrix $\mathbf{B}$ can be obtained by computing the eigenvalue problem of $(\mathbf{S}-\mathbf{F})\mathbf{K}^T\mathbf{B} = \lambda\mathbf{B}$.
Let eigenvalues be λ_i and their corresponding eigenvectors be $\mathbf{b}_i$. Assume that $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_m$.

3). Return $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_d]$

III. Hidden Space Discriminant Neighborhood Embedding

DNE is a linear feature transform so that it does not work well for the linearly non-separable data. Although the nonlinear version of DNE, or KDNE, has been proposed, its adjacent graph is still construed in the original space. Usually, the local relationship between samples in the original

space cannot be guaranteed in the high-dimensional space obtained by nonlinear mapping. Taking into account these shortcomings in DNE and KDNE, we propose a hidden space discriminant neighborhood embedding method.

A. Hidden Space

Hidden space is derived from neural networks, and is introduced to support vector machines (SVMs) in [19]. Generally, SVMs require the Mercer kernel functions. However, Nonlinear hidden functions could be any kernel ones. Some learning algorithms have been extended into the hidden space such as PCA [23] and LDA [24].

With the help of some nonlinear hidden function, data being linearly non-separable in the original space can be mapped into a high-dimensional space in which data are now linearly separable. Given N samples $\{\mathbf{x}_i\}_{i=1}^N$, we map them into a hidden space by using a hidden function $\varphi(\mathbf{x})$. Let $\mathbf{z} = \varphi(\mathbf{x})$, where $\mathbf{z}$ is the image of $\mathbf{x}$. We take kernel functions as hidden functions, and we have

$$\mathbf{z} = [k(\mathbf{x}, \mathbf{x}_1), ..., k(\mathbf{x}, \mathbf{x}_N)]^T \tag{4}$$

In hidden functions, we require only the symmetry for kernel functions instead of Mercer's condition. In addition, we can obtain the mapped samples $\mathbf{z}$. So, it is very convenient to calculate statistics for samples.

B. HDNE

In a classification task, assume that the set of labeled training samples is $\{\mathbf{x}_i, y_i\}_{i=1}^N$, where $\mathbf{x}_i \in R^m$, and $y_i \in \{1, 2, ..., c\}$. By employing the hidden function (4), the images of $\mathbf{x}_i$ in the hidden space are

$$\mathbf{z}_i = [k(\mathbf{x}_i, \mathbf{x}_1), ..., k(\mathbf{x}_i, \mathbf{x}_N)]^T$$

In the hidden space, the training set can be represented as $\{\mathbf{z}_i, y_i\}_{i=1}^N$ where $\mathbf{z}_i \in R^N$.

Since the concrete form of samples has been known in the hidden space, so we are able to directly build the adjacent graph $\mathbf{F}$ in this space. The entries of ith row and jth column in $\mathbf{F}$ is

$$F_{ij} = \begin{cases} +1, & \mathbf{z}_i \textit{ and } \mathbf{z}_j \textit{ are neighbors and } y_i = y_j \\ -1, & \mathbf{z}_i \textit{ and } \mathbf{z}_j \textit{ are neighbors and } y_i \neq y_j \\ 0, & \textit{otherwise} \end{cases} \tag{5}$$

Recall that the dimensionality of hidden space may be very high, so it needs to reduce dimensionality for efficient computation. Let the transformation matrix be $\mathbf{P} \in R^{N \times d}$, where d denotes the dimensionality of discriminant subspace. In the discriminant subspace, the sample $\mathbf{z}_i \in R^m$ is transformed to be $\mathbf{P}^T\mathbf{z}_i \in R^d$.

Let $\phi(\mathbf{P})$ and $\varphi(\mathbf{P})$ be the within-class and the between-class neighborhood scatters, respectively. The within-class neighborhood scatter $\phi(\mathbf{P})$ is defined as

$$\phi(\mathbf{P}) = \sum_{i,j,y_i=y_j} ||\mathbf{P}^T\mathbf{z}_i - \mathbf{P}^T\mathbf{z}_j||^2 \tag{6}$$

where $\mathbf{z}_i$ and $\mathbf{z}_j$ are neighbors and belong to the same class. The between-class neighborhood scatter $\varphi(\mathbf{P})$ is defined as

$$\varphi(\mathbf{P}) = \sum_{i,j,y_i \neq y_j} ||\mathbf{P}^T\mathbf{z}_i - \mathbf{P}^T\mathbf{z}_j||^2 \tag{7}$$

where $\mathbf{z}_i$ and $\mathbf{z}_j$ are still neighbors but belong to the different classes. We hope that if neighbors belong to the same class they should be close to each other in the discriminant subspace; otherwise, be far away from each other. We can implement our demand by minimizing the $\phi(\mathbf{P})$ and maximizing $\varphi(\mathbf{P})$ at the same time, which could be described as

$$\min_{\mathbf{P}} \quad \Delta(\mathbf{P}) = \phi(\mathbf{P}) - \varphi(\mathbf{P}) \tag{8}$$

Substituting (5) into the expression of $\Delta(\mathbf{P})$, we can rewritten it as follows:

$$\begin{aligned} \Delta(\mathbf{P}) &= \sum_{i,j=1}^{N} ||\mathbf{P}^T\mathbf{z}_i - \mathbf{P}^T\mathbf{z}_j||^2 F_{ij} \\ &= 2\sum_{i,j=1}^{N} (\mathbf{z}_i{}^T\mathbf{P}\mathbf{P}^T\mathbf{z}_i - \mathbf{z}_i{}^T\mathbf{P}\mathbf{P}^T\mathbf{x}_j)F_{ij} \\ &= 2\sum_{i,j=1}^{N} tr((\mathbf{P}^T\mathbf{z}_i\mathbf{z}_i{}^T\mathbf{P} - \mathbf{P}^T\mathbf{z}_j\mathbf{z}_i{}^T\mathbf{P})F_{ij}) \\ &= 2tr(\sum_{i,j=1}^{N} (\mathbf{P}^T\mathbf{z}_iF_{ij}\mathbf{z}_i{}^T\mathbf{P} - \mathbf{P}^T\mathbf{z}_jF_{ij}\mathbf{z}_i{}^T\mathbf{P})) \\ &= 2tr(\mathbf{P}^T\mathbf{Z}\mathbf{S}\mathbf{Z}\mathbf{P} - \mathbf{P}^T\mathbf{Z}\mathbf{F}\mathbf{Z}\mathbf{P}) \\ &= 2tr(\mathbf{P}^T\mathbf{Z}(\mathbf{S} - \mathbf{F})\mathbf{Z}^T\mathbf{P}) \end{aligned} \tag{9}$$

where $tr(\cdot)$ denotes the trace of a matrix, and $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_N]$, and $\mathbf{S}$ is a diagonal matrix with $S_{ii} = \sum_{j=1}^N F_{ij}$. As a result, the problem (8) can be rewritten as

$$\begin{aligned} \min_{\mathbf{P}} \quad & tr(\mathbf{P}^T\mathbf{Z}(\mathbf{S} - \mathbf{F})\mathbf{Z}^T\mathbf{P}) \\ \text{s.t.} \quad & \mathbf{P}^T\mathbf{P} = \mathbf{I} \end{aligned} \tag{10}$$

and where $\mathbf{I}$ is the identity matrix.

In the following, we introduce a lemma in [25] and give a theorem which describes the solution to (10).

Lemma 1: Suppose $\mathbf{A} \in R^{N \times N}$ is a real symmetric matrix and its minimum eigenvalue is λ_1. The solution to the minimization problem of $\boldsymbol{\eta}^T\mathbf{A}\boldsymbol{\eta}$ which subjects to $\boldsymbol{\eta}^T\boldsymbol{\eta} = 1$ and $\boldsymbol{\eta} \in R^N$ is the eigenvector corresponding to the eigenvalue λ_1.

Theorem 2: Assume that the eigenvalues of the matrix $\mathbf{Z}(\mathbf{S} - \mathbf{F})\mathbf{Z}^T$ are $\lambda_1 \leq ... \leq \lambda_{i-1} \leq \lambda_i \leq ... \leq \lambda_N$, and $\boldsymbol{\xi}_i$ is the corresponding eigenvector of eigenvalue λ_i. Then optimal $\mathbf{P}$ to the minimization problem $tr(\mathbf{P}^T\mathbf{Z}(\mathbf{S} - \mathbf{F})\mathbf{Z}^T\mathbf{P})$ is the corresponding eigenvectors of the first d eigenvalues. Namely, $\mathbf{P} = [\boldsymbol{\xi}_1, \cdots, \boldsymbol{\xi}_d]$.

Proof: Since $(\mathbf{S} - \mathbf{F})$ is a real symmetric matrix, $\mathbf{Z}(\mathbf{S} - \mathbf{F})\mathbf{Z}^T$ is also a real symmetric matrix. According to Lemma 1, if $d = 1$, only when $\mathbf{P}$ is the eigenvector corresponding to the minimum eigenvalue λ_1 of matrix $\mathbf{Z}(\mathbf{S} - \mathbf{F})\mathbf{Z}^T$ (namely $\mathbf{P} = \boldsymbol{\xi}_1$) is $\mathbf{P}^T\mathbf{Z}(\mathbf{S} - \mathbf{F})\mathbf{Z}^T\mathbf{P}$ minimum. Right now, λ_1 is the optimal value of $tr(\mathbf{P}^T\mathbf{Z}(\mathbf{S} - \mathbf{F})\mathbf{Z}^T\mathbf{P})$.

Similarly, if let $\mathbf{P}$ represent eigenvectors corresponding to the first d minimum eigenvalues (namely $\mathbf{P} = [\xi_1, \cdots, \xi_d]$), then we have $tr(\mathbf{P}^T\mathbf{Z}(\mathbf{S} - \mathbf{F})\mathbf{Z}^T\mathbf{P}) = \sum_{i=1}^{d} \lambda_i$. Right now,

$tr(\mathbf{P}^T\mathbf{Z}(\mathbf{S}-\mathbf{F})\mathbf{Z}^T\mathbf{P})$ achieves its optimal value. This completes the proof.

Theorem 2 shows that minimizing $tr(\mathbf{P}^T\mathbf{Z}(\mathbf{S}-\mathbf{F})\mathbf{Z}^T\mathbf{P})$ is equivalent to eigendecompose on the matrix $\mathbf{Z}(\mathbf{S}-\mathbf{F})\mathbf{Z}^T$. If the projection matrix is composed of the eigenvectors corresponding to the first d eigenvalues, the value of object function with respect to optimization problem (11) is minimum.

C. Comparison of DNE, KDNE and HDNE

By constructing the adjacent graph for samples in the original space, DNE makes their local structure be preserved in a discriminant subspace. In addition, the problem of finding the projection matrix is equivalent eigendecompose $\mathbf{X}(\mathbf{S}-\mathbf{F})\mathbf{X}^T$.

Nevertheless, DNE is linear so that it cannot be applied to the linearly non-separable data in the original space. As a remedy for this drawback, KDNE makes DNE extend to be nonlinear, but it still utilizes the adjacent graph constructed in the original space. For KDNE, the matrix that needs eigendecomposition is $(\mathbf{S}-\mathbf{F})\mathbf{K}^T$.

The method proposed here is first to map the data in the original space into into the hidden space and then to construct the adjacent graph in this space. The local structure of samples in the hidden space can be preserved when performing dimensionality reduction. Thus, HDNE remedies the drawbacks that DNE is not fit for nonlinear problems and KDNE cannot preserve the local structure of high-dimensional space. HDNE tries to eigendecompose $\mathbf{Z}(\mathbf{S}-\mathbf{F})\mathbf{Z}^T$.

From the above, the three methods are all based on eigendecomposition of some matrix, and then obtain the projection matrix composed of the eigenvectors corresponding to the first d minimum eigenvalues.

IV. Simulation Experiments

In this section, to validate the efficiency of HDNE, we compare it with other methods, including PCA, LDA, LPP, NPE, DNE, KFDA and KDNE on image classification problems. Here, we consider two kinds of images: face and handwritten digit.

For KDNE, KFDA and HDNE, Gaussian kernel $k(\mathbf{x},\mathbf{x}') = \exp\{-p\|\mathbf{x}-\mathbf{x}'\|\}$ with the kernel parameter $p>0$ is used. This parameter p is selected by using 5-fold cross validation.

A. Face Recognition

Consider the widely studied ORL Face dataset (from the website: http://archive.ics.uci.edu/ml/datasets.html),which is created by the University of Cambridge and has a total of 400 face images with different illumination intensity and facial expression etc., each 112 x 92, from 40 persons and 10 for each one. It also gives considerations to race, gender and facial expression and is a frequently-used face dataset.

In the face recognition experiment, we mainly focus on the effect of the dimensionality of discriminant subspace on recognition rates under different choices for K, where K is the parameter of the nearest neighbor (NN) classifier. Thus, without prior knowledge, K is set to be 1, 3 and 5 respectively. In the experiment, we randomly select 5 samples from the same person for training and the rest are for test. There are 200 training and test samples, respectively. All samples are divided by 255 to implement normalization.

For the high-dimensional original images, we first utilize PCA to reduce dimensionality from 10,304 to 100. In doing so, there are two benefits. On the one hand, computations are greatly reduced. On the other hand, the majority of noises are diminished. We repeat our experiment 100 trials and report the average result on test sets.

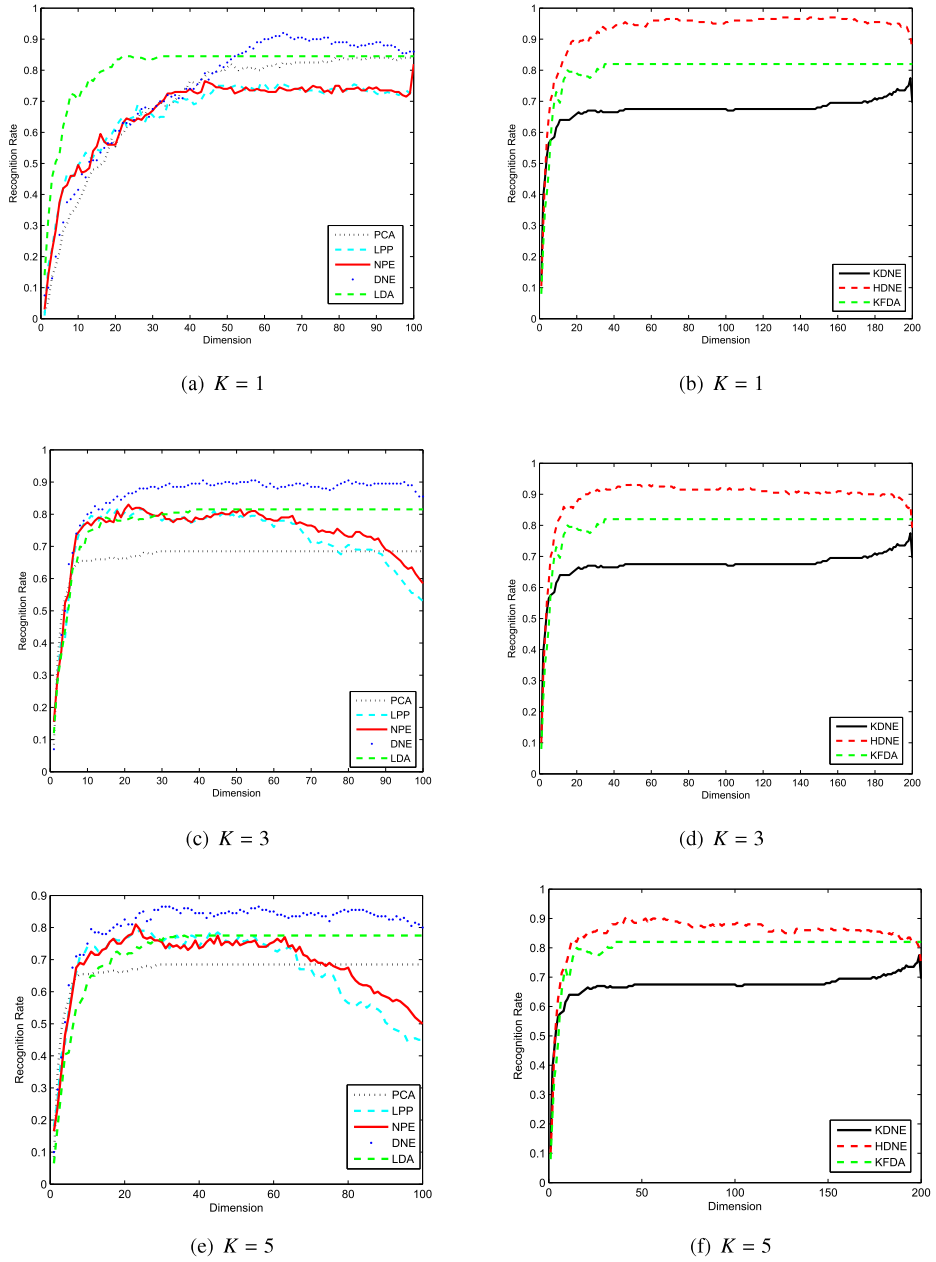
For PCA, LDA, LPP, NPE and DNE, their dimensionalities of discriminant subspace and recognition rates are plotted in one figure because their maximal dimensionalities are all 100 after dimensionality reduction. While for KDNE, KFDA and HDNE, their results are plotted in another one figure because they have the same maximal dimensionality of N.

When different K value is selected, Fig. 1 presents the corresponding performance along with the change of dimensionality for these methods. From Figs. 1, we can know that, for all the methods, at the beginning the performance improves all the time along with increasing dimensionality, and then it tends to be invariable or decreasing. From Figs. 1(a), 1(c) and 1(e), we can know that with different K values, DNE method is always able to reach a maximum, being better than PCA, LDA, LPP and NPE, in different discriminant subspace. As the nonlinear version of DNE, from Fig. 1(b), 1(d) and 1(f), KDNE,KFDA and HDNE have great change in recognition rate along with the change of dimensionality. Obviously, HDNE works better than KDNE and KFDA no matter which K is selected in our experiment.

Table I. Performance comparisons on ORL dataset (K = 1)

Method	Dimensional of subspace	Recognition rate
PCA	79	84.00
LDA	22	84.50
LPP	59	76.00
NPE	100	82.00
DNE	65	92.00
KDNE	194	79.50
KFDA	35	82.00
HDNE	71	96.50

From Fig. 1, we can see that all methods have better performance when K = 1. The larger K does not mean better since ORL face data are insufficient. Table 1 provides the best recognition rates obtained by all methods and the corresponding dimensionality of discriminant subspace for K = 1. Compared with KDNE, HDNE always has a better recognition rate and a lower dimensionality of discriminant subspace. As a result, our view that although being also a nonlinear extension of DNE, KDNE does not employ the adjacent graph to preserve the local structure so that lead to an unsatisfied recognition rate is verified. By contrast, HDNE method achieves this, which makes the following view be persuasive: compared with the practice that use kernel as nonlinear extension, this method that let the data map into high-dimensional space, and then construct adjacent graph to preserve the relationships between neighbors can work better and get a higher recognition rate in discriminant subspace.



(a) $K = 1$ (b) $K = 1$

(c) $K = 3$ (d) $K = 3$

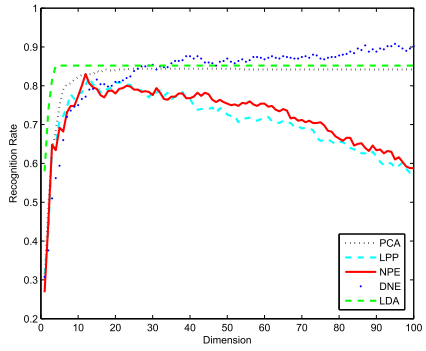
(e) $K = 5$ (f) $K = 5$

Figure 1. Recognition vs. dimensionality on ORL face dataset

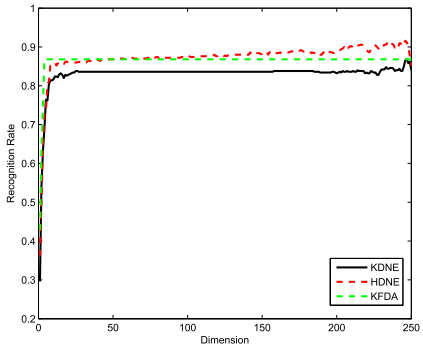
B. Handwritten Digit Recognition

Consider the MNIST dataset *(from website http://archive.ics.uci.edu/ml/datasets.html)*, which has a total of 60,000 training and 10,000 test images with total 10 classes. Five classes, including digits 1, 3, 7, 8 and 9, are selected. For each class, we randomly select 50, 100 and 150 samples from the original training set as our training set, 100 samples from the original test set as our test set. In this experiment, PCA is stilled utilized to preprocess in order to obtain the *100-dimensional* data, and nearest neighbor is selected as the classifier.

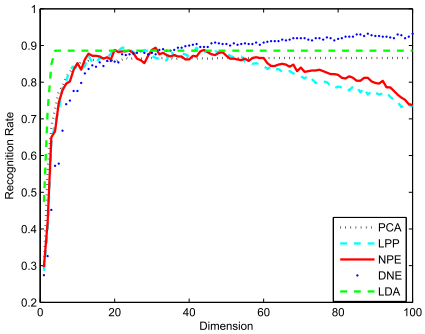
In this experiment, we will mainly perform analysis on the effect of the number of samples on the dimensionality of discriminant subspace and recognition rates. Fig. 2 clearly shows that the larger the number of samples is, the higher recognition rate is under condition of the same dimensionality for discriminant subspace. For Figs. 2(a), 2(c) and 2(e), the statuses of recognition rates for four methods are clearly presented along with the change of dimensionality of discriminant subspace. With the increase of dimensionality, the recognition rate of each method improves on the whole. Figs. 2(b), 2(d) and 2(f) respectively describe the recognition rates of KDNE and HDNE with respect to the dimensionality of discriminant subspace under condition of the same number of samples. We have the conclusion that with the same number of samples and the same dimensionality of discriminant subspace, as to recognition rate, HDNE obviously works better than KDNE and KFDA.



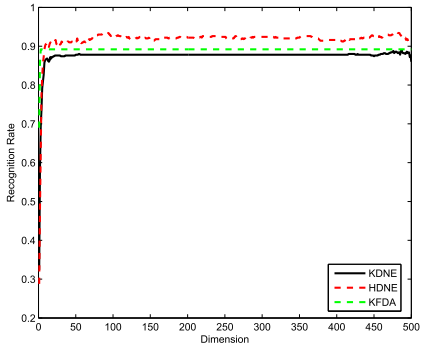
(a) 50 training samples



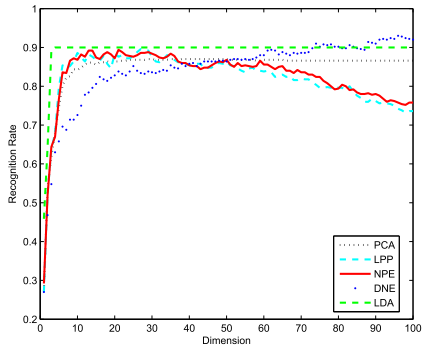
(b) 50 training samples



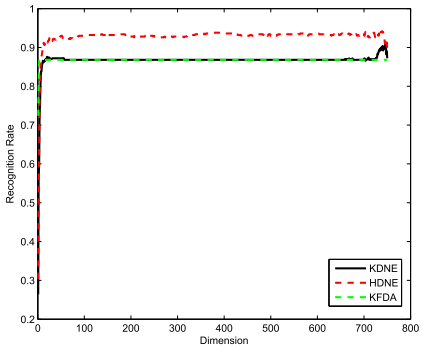
(c) 100 training samples



(d) 100 training samples



(e) 150 training samples



(f) 150 training samples

Figure 2. Recognition vs. dimensionality on MNIST dataset

Table II. PERFORMANCE COMPARISONS ON THE MNIST DATASET (50 *trainingsamples*)

Method	Dimensional of subspace	Recognition rate
PCA	24	84.40
LPP	14	82.20
NPE	12	83.00
LDA	4	86.40
DNE	95	90.10
KDNE	247	86.80
KFDA	3	86.86
HDNE	244	91.80

Table III. PERFORMANCE COMPARISONS ON THE MNIST DATASET (100 *trainingsamples*)

Method	Dimensional of subspace	Recognition rate
PCA	26	86.80
LPP	22	89.20
NPE	31	89.40
LDA	4	87.80
DNE	88	91.60
KDNE	487	89.00
KFDA	3	89.20
HDNE	480	93.80

Tables 2-4 show the optimal recognition rates for each method in the whole discriminant subspace with the certain number of samples, from which the conclusion is that recognition rate of HDNE is higher than the other methods. As a nonlinear extension of DNE as well, HDNE also has a higher recognition rate than KFDA and KDNE.

Table IV. PERFORMANCE COMPARISONS ON THE MNIST DATASET (150 *trainingsamples*)

Method	Dimensional of subspace	Recognition rate
PCA	29	87.00
LPP	27	89.40
NPE	21	89.40
LDA	4	90.00
DNE	96	91.80
KDNE	746	90.60
KFDA	3	86.98
HDNE	739	94.40

V. CONCLUSIONS

HDNE is proposed by introducing hidden functions, which is a nonlinear extension of DNE. Specifically, it performs analysis on the preserved local structure in hidden space. The data being linearly non-separable in original space are linearly separable in hidden space and at the same time the adjacent graph is constructed to preserve the local structure of data. From experimental results on ORL dataset with different K values in K-nearest neighbor and on MNIST dataset with the different number of samples, we have the conclusion that HDNE has a better performance than the other methods.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 61373093 and 61033013, by the Natural Science Foundation of Jiangsu Province of China under Grant Nos. BK2011284 and BK201222725, by the Natural Science Foundation of the Jiangsu Higher Education Institutions of China under Grant No.13KJA520001,and by the Qing Lan Project.

REFERENCES

[1] J. B. Tenenbaum, V. de Silva and J. C. Langford, A global geometric framework for nonlinear dimensionality reduction, *Science* 290, 2319-2323, 2000.

[2] S. Roweis and L. K. Saul, Nonlinear dimensionality reduction by locally linear embedding, *Science*, 290, pp. 2323-2326, 2000.

[3] S. Roweis and L. K. Saul, Nonlinear Dimensionality Reduction by Locally Linear Embedding, *Science*, vol. 290, 22 December, 2000.

[4] X. F. He and P. Niyogi, Locality Preserving Projections, *Advances in Neural Information Processing Systems 16* Vancouver, British Columbia, Canada, 2003.

[5] X. F. He, D. Cai, S. C. Yan, eta, Neighborhood Preserving Embedding, *Proceeding of the IEEE International Conference on Computer Vision.* Beijing, China, pp, 1208-1213, 2005

[6] H. T. Chen, H. W. Chang and T. L. Liu, Local discriminant embedding and its variants, *Proceedings of International Conference on Computer Vision and Pattern Recognition*, 2005.

[7] M. Belkin and P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, *Neural Computation*, vol. 15, pp. 1373-1396, 2003.

[8] X. He, S. Yan, Y. Hu and H. J. Zhang, Learning a Locality Preserving Subspace for Visual Recognition, *Proc. 9th International Conference on Computer Vision*, France, 2003.

[9] K. C. Lee, J. Ho, M. H. Yang and D. Kriegman. Video-Based Face Recognition Using Probabilistic Appearance Manifolds, *IEEE Conference on Computer Vision and Pattern Recognition*, Madison, Wisconsin, 2003.

[10] B. Scholkopf, A. Smola and K. R. Muller, Nonlinear Component Analysis as a Kernel Eigenvalue Problem, *Neural Computation*, *10* (5)*:* 1299-1319, 1998.

[11] M. Sugiyama, Local Fisher Discriminant Analysis for Supervised Dimensionality Reduction, *Proc of the 23rd International Conference on Machine Learning*,Pittsburgh, USA, pp, 905-912. 2006.

[12] P. N. Belhumeur, J. P. Hespanha and D. J. Kriegman, Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection, PAMI, vol. 19, no. 7, pp. 711-720, July. 1997.

[13] M. Belkin and P. Niyogi, Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering, NIPS 14, 2001.

[14] C. Hu, Chang Y, R. Feris and M. Turk, Manifold Based Analysis of Facial Expression, *IEEE Workshop on Face Processing in Video*,2004.

[15] M. H. Yang, Kernel Eigenfaces vs. Kernel Fisherfaces : Face Recognition Using Kernel Methods, AFGR, pp. 205-211, 2002.

[16] W. Zhang, X. Y. Xue, H. Lu and Y. F. Guo, Discriminant Neighborhood Embedding for Classification, *Pattern Recognition*, *39* (11)*:* 2240-2243, Nov. 2006.

[17] S. Saitoh, Theory of Reproducing Kernels and Its Applications, *UK* : *Longman Scienti fic and Technical*, 2004.

[18] J. Shawe−Taylor and N. Cristianini. Kernel Methods for Pattern Analysis, *Cambridge University Press*, 2004.

[19] L. Zhang, W. D. Zhou and L. C. Jiao, Hidden Space Support Vector Machines IEEE Trans, vol. 15, no. 6, 2004.

[20] K. Fukunaga, *Introduction to Statistical Pattern Recognition,* 2nd ed. New York : Academic, 1991.

[21] K. Mtiller, S. Mika, G. Riitsch, K. Tsuda and B. Sch61kopf, An Introduction to kernel-based learning algorithms, *IEEE Transactions on Neural Networks*, Vol. 12, pp. 181−201, 2001.

[22] D. M. J. Tax and R. P. W. Duin, Support vector data description, Mach. Learn. 54 (2004) 45-66

[23] W. D. Zhou, L. Zhang and L. C. Zhang. Hidden space principal component analysis 10*th* Pacific, *Asia Conference on Knowledge Discovery and Data Mining*, LNAI 3981, 801 − 805, 2006.

[24] L. Zhang, W. D. Zhou and P. C. Chang. Generalized Nonlinear Discriminant Analysis and Its Small Sample Size Problems Neurocomputing, 74, 568 − 574, 2011.

[25] W. Zhang and X. Y. Xue. *Study on Feature Transformation Algorithm based on K − Nearest − Neighbor Classification Rule,* Fudan University, China, 37-40, 2007.