Pattern Recognition ▮▮▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Pattern Recognition

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

Weighted locally linear embedding for dimension reduction

Yaozhang Pan, Shuzhi Sam Ge*, Abdullah Al Mamun

Social Robotics Lab, Interactive Digital Media Institute, and Department of Electrical & Computer Engineering, National University of Singapore, Singapore 119077, Singapore

ARTICLE INFO

Article history:
Received 3 August 2007
Received in revised form 27 May 2008
Accepted 20 August 2008

Keywords:
Nonlinear dimensionality reduction
Manifold learning
Feature extraction
Locally linear embedding

ABSTRACT

The low-dimensional representation of high-dimensional data and the concise description of its intrinsic structures are central problems in data analysis. In this paper, an unsupervised learning algorithm called weighted locally linear embedding (WLLE) is presented to discover the intrinsic structures of data, such as neighborhood relationships, global distributions and clustering. The WLLE algorithm is motivated by locally linear embedding (LLE) algorithm and cam weighted distance, a novel distance measure which usually gives a deflective cam contours for equal-distance contour in classification for an improved classification. It is a major advantage of the WLLE to optimize the process of intrinsic structure discovery by avoiding unreasonable neighbor searching, and at the same time, allow the discovery adapt to the characteristics of input data set. Furthermore, the algorithm discovers intrinsic structures which can be used to compute manipulative embedding for potential classification and recognition purposes, thus can work as a feature extraction algorithm. Simulation studies demonstrate that the WLLE can give better results in manifold learning and dimension reduction than LLE and neighborhood linear embedding (NLE), and is more robust to parameter changes. Experiments on face images data sets and comparison to other famous face recognition methods such as kernel-PCA (KPCA) and kernel direct discriminant analysis (KDDA) are done to show the potential of WLLE for real world problem.

1. Introduction

Many problems in machine learning begin with the preprocessing of raw high-dimensional signals, such as face images, speech spectrograms, EEG and ECG signals for medical diagnose. For convenience of subsequent operations such as classification [1], image processing [2] and outlier detection [3], the preprocessing should extracts and highlights the inherent properties hidden in the high-dimensional observations and represents the intrinsic structures in a more compact and efficient way. However, the representations must be learned or discovered automatically in the case that no prior knowledge about the data is known. Automatic methods which discover hidden structures from the statistical regularities of large data sets can be studied in the general framework of unsupervised learning [4,5].

The strategies and methodologies to solve this problem can be categorized into linear and nonlinear methods. Principal component analysis (PCA) is a linear projection method that emphasizes on the features of observations with large variability that can be discovered using cross correlation [6]. Classical multidimensional scaling (MDS) seeks to preserve pairwise distance and simple formations of observations such as triangle [7]. Due to the nonlinear relationships of high-dimensional observations in nature, several geometry-oriented methods are introduced by mapping high-dimensional inputs into low-dimensional embeddings nonlinearly such as ISOMAP, by which the geodesic relationship among the input data and the calculated low-dimensional embeddings remains consistent [8].

Linear techniques based on PCA or linear discriminant analysis (LDA) cannot provide reliable and robust solutions to nonlinearity distribution such as face patterns. As a result, nonlinear techniques such as kernel-PCA (KPCA) [9], generalized discriminant analysis (GDA), and kernel direct discriminant analysis (KDDA) [2] was proposed to solve the problem of nonlinearity in data distribution. The KPCA was proposed in Ref. [9], which is as simple as standard PCA because no nonlinear optimization is involved. However, it may have trouble when very large number of observations is needed to be processed. KDDA is proposed and used to do face recognition to deal with the nonlinearity and small sample size of the face pattern's distribution and data sets [2]. All of these kernel-based algorithms have a further problem of large training sample, overfitting, and finding suitable kernel functions for each specific data set.

Local linear embedding (LLE) is an unsupervised learning algorithm that computes low-dimensional, neighborhood-preserved embeddings of high-dimensional inputs [4,5]. Unlike clustering methods for local dimension reduction [10], LLE maps its inputs into

* Corresponding author. Tel.: +65 6516 6821; fax: +65 6779 1103.
E-mail addresses: yaozhang.pan@nus.edu.sg (Y. Pan), samge@nus.edu.sg (S.S. Ge), eleaam@nus.edu.sg (A. Al Mamun).

0031-3203/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2008.08.024

a single global coordinate system of lower dimensionality, and its optimizations do not involve local minima [4]. By exploiting the local symmetries of linear reconstructions, LLE is able to learn the global structure of nonlinear manifolds. It eliminates the need to estimate pairwise distances between widely separated data points and recovers global nonlinear structure from locally linear fits [5]. It is suitable to solve the problem of dimension reduction arises in many fields of information processing, including machine learning, data compression, scientific visualization, pattern recognition, and neural computing [11–13].

However, LLE algorithm, as well as many other machine learning and pattern recognition algorithms, such as nearest neighbor classifier [14,15], radial basis function networks [16], support vector machines (SVMs) for classification [17], k-means algorithm for clustering [18,19], actually rely on a distance metric. As a direct result, the performance of the method depends critically on the choice of appropriate metric. Many early works have been carried out to relax this restriction, such as optimal metric for k-nearest neighbor density estimation [15], optimal local metric [14] and optimal global metric [20]. More recent research along this line continued to develop various locally adaptive metrics [20–25] for metric learning-based algorithms; in Ref. [26], how to find a better distance measure for similarity estimation was discussed, and a group of new distance measures are derived and proved to be more efficient in feature extraction than traditional Euclidean and Manhattan distances; in Ref. [27], the authors extended the LLE procedure with a weighting scheme by associating weights with face images to represent their probability of occurrence, and obtained better performance on face recognition.

The existing methods handle this problem only from the aspect of the query point. They analyze the measurement space emanating from the query point, and study how the distance measure should be changed or weighted. These approaches only examine a small local region surrounding the query sample, as such the most of the inter-prototype information is neglected. To solve this problem, cam weighted distance for improving nearest neighbor finding is developed in Ref. [28]. The "cam weighted distance" was so named because that it usually gives a deflective cam contours for equal-distance contour in classification as mentioned and shown in Ref. [28, Fig. 1]. This method optimizes the distance measure with respect to the analysis of the inter-prototype relations. Since the prototypes are not isolated instances, the nearby prototypes actively affect the confidence level of the information provided by the prototype being considered. As a result, to improve distance measure globally, we should consider both variances with its own orientation and discrimination with respect to its different surroundings of each prototype.

In this paper, we proposed weighted locally linear embedding (WLLE) by modifying the LLE algorithm based on the weighted distance measurement to improve the dimension reduction and internal feature extraction performance especially for the deformed distributed data. In the case that data distribution is deformed because of the attraction, repulsion, strengthening effect and weakening effect each sample point receives from its neighbors, Euclidean distance for measuring the similarity will lead to performance decline. By taking into account the distribution information surrounding each prototype to optimize the distance measure, we can improve the neighbor finding procedure of LLE algorithm and avoid the redundancy and overlapping due to improper neighbor selection. Better neighbors selection will make the dimension reduced representations more accurate to represent the internal feature, characteristic, and structure of the high-dimensional data.

The main contributions of the paper are as follows:

(i) a novel weighted distance measurement for neighborhood searching is adopted to solve the problem of neighbors redundancy and overlapping when the samples are not well-distributed;
(ii) a modified LLE based on the weighted distance measurement called WLLE is presented to give better performance of internal feature extraction;
(iii) the problem that the LLE cannot give faithful embeddings for a kind of difficult data set, in which some data are noisy, sparse or weakly connected, is solved by WLLE; and
(iv) the WLLE algorithm is tested using several manifolds and images, the results of simulations demonstrate that the WLLE not only has better performance in manifold learning, but also is more robust to parameter changes.

The rest of the paper is organized as follows. The main features of LLE algorithm and modified NLE algorithm are briefly introduced in Section 2. Then weighted distance measurement is introduced in Section 3 for modification of LLE in the following sections. In Section 4, weighted distance measurement is adopted to form a new nonlinear dimension reduction algorithm, WLLE. Simulation studies on both artificial manifold data sets and real-world data sets are given in Section 5. Then, the motivation and origin of this work and the main advantages of WLLE are discussed in Section 6. Section 7 concludes this work at last.

2. LLE and NLE

For ease of the forthcoming discussion, we firstly introduce the main features of the LLE algorithm. It is an unsupervised learning algorithm that attempts to map high-dimensional data to low-dimensional, neighborhood-preserved embeddings. It is based on the simple geometric intuitions: (i) each high-dimensional data point and its neighbors lie on or close to a locally linear patch of a manifold [4], and (ii) the local geometric characterization in original data space is unchanged in the output data space. From a mathematic point of view, the problem LLE attempts to resolve is: given a set $X=[x_1,x_2,\ldots,x_N]$, where x_i $(i=1,\ldots,N)$ is ith node on a high-dimensional manifold embedded in R^D, i.e., $x_i \in R^D$, and then find a set $Y=[y_1,y_2,\ldots,y_N]$ in R^d, where $d \ll D$ such that the intrinsic structure in X can be represented by that of Y.

The neighbor finding process of LLE is usually carried out using the grouping technique such as k-nearest neighbors (KNN) or choosing neighbors within a ball of fixed radius (ε-neighborhoods) based on Euclidean distance for each data point in the given data set. These neighbors are then used to reconstruct the given point by linear coefficients. The KNN method is widely used due to its simplicity and ease of implementation. However, due to the complexity, nonlinearity and variety of high-dimensional input samples, the K is difficult to choose properly to obtain a acceptable level of redundancy and overlapping. A small K leads to possible isolation of nodes. For the extreme case where $K=0$, all nodes are totally separated and no intrinsic structure can be observed. A large K increases redundancy and overlapping. For instance, if $K=N-1$, each individual node is directly connected to the rest of the nodes such that all nodes belongs to one cluster, no matter what the exact number of clusters is.

Therefore, the choice of K affects the tradeoff between the redundancy present in the structure and the number of isolated nodes. Thus, an adaptive scheme to select K is more appropriate for finding neighbors. Based on this idea, a modified algorithm named NLE was proposed [29,30]. It is an adaptive scheme that select neighbors according to the inherent properties of the input data substructures. By defining d_{ij} the Euclidean distance from mode x_j to x_i

Please cite this article as: Y. Pan, et al., Weighted locally linear embedding for dimension reduction, Pattern Recognition (2008), doi: 10.1016/j.patcog.2008.08.024

and S_i the data set containing all the neighbor of x_i, the neighbor finding procedure of NLE for a given point x_i can be summarized as follows:

(i) If $d_{ij} = \min\{d_{im}\}$, $\forall m \in 1, 2, \ldots, N$, then x_j is regarded as a neighbor of the node x_i, we initial $S_i = \{x_j\}$.
(ii) If $d_{ik} = 2_{ed} \min\{d_{im}\}$, $\forall m \in 1, 2, \ldots, N$, then x_k is regard as a neighbor of node x_i if $d_{jk} > d_{ik}$.
(iii) When S_i contains two or more elements, for $\forall m \in S_i$, if $d_{jm} > d_{ji}$ and $d_{jm} > dmi$ are satisfy, then $S_i = S_i \cup \{x_m\}$.

This modified adaptive neighbor searching algorithm not only solves the problem of redundancy and overlapping but also avoids the trial and error operation, which is a obvious shortcoming of KNN algorithm. However, according to NLEs neighborhood selection criterion, the number of neighbor selected to be used is usually small. Generally speaking, for a data point x, the NLE determine the nearest neighbor x_1 to be the first neighbor point; from the second nearest neighbor, say, x_2, it will be considered a neighbor of x only when the distance between x_2 and x_1 is larger than the distance between x_2 and x. As a result, it is not surprising that the number of neighbors selected by NLE algorithm is usually much smaller than other algorithms. For example, according to the experiment on two peaks data sample, the average number of neighbor for 1000 samples chosen by NLE is only 3.74 [31]. This may result from the strict neighbor selection rules of NLE. Besides this experiment, we have also do many other simulations and find the neighbor size by NLE algorithm is smaller than other ones. In that case, the reconstruction information may not be enough for data reconstruction.

After carefully considering the LLE and NLEs neighbor selection criterion, we propose a new algorithm by using weighted distance measurement in neighbor searching. The new algorithm can solve the problem of redundancy in LLE and avoid NLEs problem that no enough data are chosen as neighbors at the same time.

3. Distribution deformation and weighted distance

In the data manipulation like nearest neighbor searching, each datum can be regarded as the center of a probability distribution and the similarity of its neighbors to the datum can be measured by Euclidean distance with the assumption that samples are well-distributed. However, because of the attraction, repulsion, strengthening effect and weakening effect between data, the standard normal distributions will be greatly deformed. Obviously, neglecting such a deformation and still using Euclidean distance to measure the similarity will lead to performance decline. As mentioned in Ref. [4], the data set should be sufficient and well-sampled, otherwise the performance of LLE algorithm will not be good enough. For example, as illustrated in Fig. 1, the samples are not well-distributed, data density changes sharply within a small area, the query point is marked by a cross, and its neighbors marked by circles. We use ε-neighborhoods algorithm to finding nearest neighbors of the query point from its neighbors. For this deformed distribution data set, ε-neighborhoods method based on standard Euclidean distance measurement selects neighbors from a single direction, and these neighbors are closely gathered. Obviously, if we use these chosen neighbors to reconstruct the query point, the information captured in this direction will have serious redundancy; at the same time, no information from other directions are reserved for query point reconstruction. These chosen neighbors cannot represent and reconstruct the query point well, most internal features and intrinsic structure will be lost after dimension reduction by LLE.

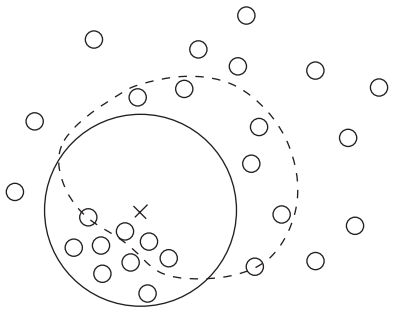

Fig. 1. Select nearest neighbors using ε-neighborhoods algorithm by Euclidean distance (solid line) and weighted distance (dash line).

To solve this problem, we introduce the weighted distance measurement motivated by Ref. [28]. The main idea of the weighted distance measurement is giving a different but appropriate distance scale to each prototype to make the distance measure more reasonable for representing the global distribution of the data set. Fig. 1 shows the advantages of this scaled adaptive distance measurement. The modified ε-neighborhoods method based on weighted distance measurement select neighbors more reasonable than the one based on standard Euclidean distance by giving the prototype data with high density a smaller weight scaling while those with low density a larger weight scaling. Thus, the previous redundancy and deficiency problem can be solved.

As defined in Ref. [28], a simple yet effective transformation to simulate the possible deformation of data distribution is constructed.

Definition 1 (*Deformed distribution*). Consider a d-dimensional random vector $Y = (Y_1, Y_2, \ldots, Y_d)^{\mathrm{T}}$ that takes a standard d-dimensional normal distribution $N(0, I)$, that is, it has a probability density function

$$f(y) = \frac{1}{(2\pi)^{d/2}} \mathrm{e}^{-1/2 y^{\mathrm{T}} y} \tag{1}$$

Let a random vector X be defined by the transformation

$$X = \left(a + b \frac{Y^{\mathrm{T}}\tau}{\|Y\|}\right) Y \tag{2}$$

where Y denotes the original well-distributed data set, $a > b \geqslant 0$ are the parameters reflect overall scale and orientation of distribution, τ is a normalized vector denoting the deformation orientation, $\|Y\| = \sqrt{Y^{\mathrm{T}}Y}$, and X represent the deformed distribution with parameters a and b in the direction τ, denoted as $X = D_d(a, b, \tau)$ [28].

According to the definition, a deformed distribution biases towards a specific direction, which makes it an eccentric distribution. Thus, the assumption that data are well-distributed is dissatisfied for Euclidean distance to describe the similarity between data points. Instead, an inverse transformation $Y = X/(a + b\cos\theta)$ is used to restore the deformation, and then we can measure the distance normally since the transformed distribution is not eccentric anymore. This is the main idea of the weighted distance measurement. This weighted distance redresses the deformation problem and should be more reasonable to evaluate the similarity for data set that is not well-distributed.

Definition 2 (*Weighted distance*). Assume that $x_0 \in R^d$ is the center of a deformed distribution $D_d(a, b, \tau)$. The weighted distance from a point $x \in R^d$ to x_0 is defined to be

$$\mathrm{Dist}(x_0, x) = \frac{\|x - x_0\|}{a + b\dfrac{(x - x_0)^{\mathrm{T}}\tau}{\|x - x_0\|}} \tag{3}$$

Please cite this article as: Y. Pan, et al., Weighted locally linear embedding for dimension reduction, Pattern Recognition (2008), doi: 10.1016/j.patcog.2008.08.024

or

$$\mathrm{Dist}(x_0,x) = \|x - x_0\|/(a + b\cos\theta) \tag{4}$$

where θ is the angle between vectors $x - x_0$ and τ, and $1/(a+b\cos\theta)$ is the weight of the distance from x to x_0 [28].

One disadvantage of the weighted distance measurement is just a weighted distance, but not a metric, since $\mathrm{Dist}(x_0,x)$ may not equal to $\mathrm{Dist}(x,x_0)$ under the definition of weighted distance. In fact, it has been discussed in Ref. [32] that non-Euclidean or non-metric measures can be informative in statistical learning algorithms.

To facilitate parameter estimation for weighted distance, we first present some properties.

Theorem 1. *If a random vector $X = D_d(a,b,\tau)$, then $E(X) = c_1 b\tau$ and $E(\|x\|) = c_2 a$, where c_1 and c_2 are constants.*

$$c_1 = \sqrt{2}\,\frac{\Gamma((d+1)/2)}{\Gamma(d/2)d} \tag{5}$$

$$c_2 = \sqrt{2}\,\frac{\Gamma((d+1)/2)}{\Gamma(d/2)} \tag{6}$$

where d is the dimensionality of the random vector X; Γ is the Gamma function $\Gamma(k) = \int_0^\infty t^{k-1}\mathrm{e}^{-t}\,\mathrm{d}t$ $(k>0)$ [28].

The expected value (or expectation) of a random variable is the sum of the probability of each possible outcome of the experiment multiplied by its value. Thus, for the random vector X, which has a deformed distribution $D_d(a,b,\tau)$, $X = D_d(a,b,\tau)$, we can calculate its expectation using the origin point of this distribution, x_i, and its k-nearest neighbors, $X_i = \{x_{i1}, x_{i2}, \ldots, x_{ik}\}$, approximately.

First, we convert X_i to a set of vectors $V_i = \{v_{i1}, v_{i2}, \ldots, v_{ik}\}$, where $v_{ij} = x_{ij} - x_i$, $j = 1,2,\ldots,k$. Then, we calculate the mean of v_{ij},

$$\hat{G}_i = \sum_{j=1}^{k} v_{ij}/k \tag{7}$$

to estimate $E(X)$, and the mean of $\|v_{ij}\|$,

$$\hat{L}_i = \sum_{j=1}^{k} \|v_{ij}\|/k \tag{8}$$

to estimate $E(\|X\|)$.

As τ is a normalized vector denoting the deformation orientation of the deformed distribution, it can be calculated as

$$\hat{\tau}_i = \frac{\hat{G}_i}{\|\hat{G}_i\|} \tag{9}$$

Since τ is a normalized vector with unity gain, according to Theorem 1, $E(X) = c_1 b\tau$ and $E(\|x\|) = c_2 a$, and with the approximation of expectation, $E(X)$ and $E(\|X\|)$, we can easily obtain the parameters a, b:

$$\hat{a}_i = \frac{\hat{L}_i}{c_2} \tag{10}$$

$$\hat{b}_i = \frac{\|\hat{G}_i\|}{c_1} \tag{11}$$

4. Weighted locally linear embedding

The weighted distance can measure the similarity more reasonable for the deformed-distributed data set than standard Euclidean distance and is suitable for many distance based methods. Accordingly, in this paper, we propose a novel dimension reduction algorithm, WLLE which use weighted distance measurement to improve the dimension reduction and internal feature extraction performance especially for the deformed distributed data.

For data points $X = \{x_1, x_2, \ldots, x_N\}$ in the high-dimensional space R^D, the goal of dimension reduction is to calculate a representative in low-dimensional space R^d for the high-dimensional data, where $d \ll D$.

We attempt to express data point number x_i as a linear combination of its k-nearest neighbors x_j, $j = 1,2,\ldots,k$.

$$\hat{x}_i = \sum_{j\in\Omega_i} w_{ij} x_j \tag{12}$$

where Ω_i is the neighborhood of sample x_i. In the original algorithm, standard Euclidean metric is used to select the nearest neighbors. However, in this work, we utilize the weighted distance measurement as in Section 3 in order to improve performance when the data set are deformed-distributed.

The optimal weight matrix w_{ij} for data reconstruction can be obtained by minimizing the approximation error const function

$$\varepsilon(W) = \sum_i \left\| x_i - \sum_{j\in\Omega_i} (w_{ij} x_j) \right\|^2 \tag{13}$$

subject to the constraints

$$j \notin \Omega_i \Rightarrow w_{ij} = 0 \tag{14}$$

$$\sum_{j\in\Omega_i} w_{ij} = 1 \tag{15}$$

where $w_i = [w_{i1}, \ldots, w_{ik}]$ are the weights connecting sample x_i to its neighbors. The first constraint says that only data points in the neighborhood of data point i should be used in the reconstruction of $\hat{x}_i$, while the second constraint imposes invariance to translation.

To calculate the optimal weights, we first rewrite the approximation error cost function (13) as

$$\begin{aligned}\varepsilon(W) &= \|x_i - \hat{x}_i\| \\ &= \left\| x_i \sum_{j\in\Omega_i} w_{ij} - \sum_{j\in\Omega_i} (w_{ij} x_j) \right\| \\ &= \sum_{j\in\Omega_i} w_{ij} \sum_{k\in\Omega_i} w_{ik} (x_i - x_j)^{\mathrm{T}} (x_i - x_k)\end{aligned} \tag{16}$$

By defining

$$C_i(j,k) = (x_i - x_j)^{\mathrm{T}} (x_i - x_k) \tag{17}$$

and applying a Lagrange multiplier η_i, the approximation error becomes

$$\varepsilon(W_i) = \sum_{j\in\Omega_i} w_{ij} \sum_{k\in\Omega_i} w_{ik} C_i(j,k) + \eta_i \left(\sum_{j\in\Omega_i} w_{ij} - 1 \right) \tag{18}$$

The optimal weights are found by requiring the partial derivatives with respect to each weight w_{ii} to be zero,

$$\frac{\partial\varepsilon(w_i)}{\partial w_{ij}} = \sum_{k\in\Omega_i} w_{ik} C_i(j,k) + \eta_i = 0, \quad \forall j \in \Omega_i \tag{19}$$

Please cite this article as: Y. Pan, et al., Weighted locally linear embedding for dimension reduction, Pattern Recognition (2008), doi: 10.1016/j.patcog.2008.08.024

The desired solution w_i is found by simply solving the equations,

$$\sum_{k\in\Omega_i} C_i(j,k)w_{ik} = 1 \tag{20}$$

and then rescale the weights so that they sum to one.

In unusual cases, it can arise that the matrix (17) is singular or nearly singular. In this case, the least square problem for finding the weights does not have a unique solution. In order to guarantee numerical stability we regulate C by

$$C_i(j,k) \leftarrow C_i(j,k) + \eta_r I \tag{21}$$

where $\eta_r \ll \mathrm{trace}(C)$ is a small constant to be defined as part of the algorithm, and I is an identical matrix.

The final step of LLE is to compute a low-dimensional embedding based on the reconstruction weights w_{ij} of the high-dimensional inputs x_i. The high-dimensional data are mapped into the low-dimensional space R^d by requiring reconstruction to work as well as possible. This leads to another minimization problem [29], the low-dimensional outputs y_i, $i=1,2,\ldots,N$ are found by minimizing the cost function,

$$\Phi(Y) = \sum_i \left\| y_i - \sum_{j\in\Omega_i} w_{ij} y_j \right\|^2 \tag{22}$$

where $Y=[y_1,\ldots,y_N]$ consist of the data points embedded into the low-dimensional space. This minimization problem is not well-posed without further constraints. Zero mean and unity covariance is used in the LLE algorithm to make the problem well-posed. In other words, $\boldsymbol{Y}$ should obey the constraints

$$\sum_{i=1}^{N} y_i = \mathbf{0} \tag{23}$$

$$\frac{1}{N} YY^{\mathrm{T}} = I \tag{24}$$

where the first constraint is to assure that coordinates y_i can be translated by a constant displacement without affecting the cost, while the second constraint imposes unit covariance of the embedding vectors.

In matrix form, the cost function can be written as

$$\begin{aligned}\Phi(Y) &= \mathrm{Tr}[(Y-YW)^{\mathrm{T}}(Y-YW)]\\ &= \mathrm{Tr}[(Y-YW)(Y-YW)^{\mathrm{T}}]\\ &= \mathrm{Tr}[Y(I-W)(I-W)^{\mathrm{T}}Y^{\mathrm{T}}]\\ &= \mathrm{Tr}[YMY^{\mathrm{T}}]\end{aligned} \tag{25}$$

where the symmetric matrix

$$M = (I-W)(I-W)^{\mathrm{T}} \tag{26}$$

The minimum of Eq. (25) subject to the constraint of Eq. (24) can be obtained by finding the d smallest eigenvectors of M. The minimal value of $\Phi(Y)$ equals the sum of the eigenvalues of M. Notice that

$$M\mathbf{1} = (I-W)(I-W)^{\mathrm{T}}\mathbf{1} = 0 \tag{27}$$

due to the requirement $\sum_{j\in\Omega_i} w_{ij} = 1$. Therefore, the smallest eigenvalue is automatically zero with corresponding eigenvector $\mathbf{1}$, here $\mathbf{1}$ is a vector in which all elements are 1. Since the eigenvectors are mutually orthogonal discarding it fulfills the constraint of Eq. (23). To summarize, the d-dimensional embedding $Y \in R^{d\times N}$ consists of eigenvector number $2,\ldots,d+1$ as its rows.

The whole procedure of dimension reduction as well as the construction of weighted distance measurement are detailed in Algorithm 1.

Algorithm 1. Weighted locally linear embedding procedure

Phase 1: Construct Weighted Distance

Given a raw high-dimensional data set $D=\{x_i\}$, $i=1,2,\ldots,N$, $x_i \in R^D$, and a parameter k_w, for an arbitrary datum $x_i \in D$,

1: Find k_w-nearest neighbors $X_i=\{x_{i1},x_{i2},\ldots,x_{ik_w}\}$, $X_i \subset D$ by compare the Euclidean distances between all neighbor points and the query point.
2: Obtain V_i with its elements to be calculated as $v_{ij}=x_{ij}-x_i$, $j=1,\ldots,k_w$.
3: Calculate $\hat{G}_i$ and $\hat{L}_i$, according to Eqs. (7) and (8).
4: Estimate a_i, b_i, τ_i by using $\hat{G}_i$ and $\hat{L}_i$, according to Eqs. (10), (11) and (9).

Phase 2: Search Neighborhood

For an arbitrary datum $x_i \in D$, $i=1,2,\ldots,N$, find k-nearest neighbors based on the weighted distance

1:Calculate the weighted distance from x_i to $\forall x_j \in D$, $j \neq i$ according to Eq. (3).
2: Find the k-nearest neighbor $X_j=\{x_{j1},x_{j2},\ldots,x_{jk}\}$, $X_j \subset D$, which satisfy

$$\mathrm{Dist}(x_j,x_i) < \mathrm{Dist}(x_k,x_i) \tag{28}$$

for $\forall x_j \in X_j$, $\forall x_i \in D$ and $\forall x_k \in D \notin X_j$.

Phase 3: Calculate Optimal Reconstruction Weights

1: Compute local covariance matrix according to Eq. (17).
2: Regulate the local covariance matrix according to Eq. (21).
3: Compute the reconstruction weights according to Eq. (20).

Phase 4: Compute Low-Dimensional Embedding

1: Construct a symmetric $N\times N$ matrix according to Eq. (26).
2: Calculate eigenvalues and eigenvectors of the symmetric matrix (26).
3: Obtain low-dimensional embedding using bottom $d+1$ eigenvectors (according to smallest $d+1$ eigenvalues) of matrix (26).

5. Experimental evaluation

To evaluate the dimension reduction and feature extraction effect of the WLLE, the results of several sets of experiments are presented in this section. The WLLE algorithm is tested on artificial manifold data sets and compared to other manifold learning algorithms such as LLE, NLE, ISOMAP and Laplacian Eigenmaps. Among these artificial manifold data sets, the Swiss roll and Toroidal helix are used to test the ability to unfold the uniformly distributed manifolds; Gaussian distribution and punched sphere are used to test the ability to unfold the no-uniformly distributed manifolds; the 3D clusters is used to test the clustering ability for all the dimension reduction algorithms.

Two real-world data sets, different subjects' faces and different poses of a face, are used to demonstrate the practical value of the algorithm we proposed. LLE, NLE, PCA, KPCA and KDDA are also applied to these real-world data sets to do comparison with WLLE. The first experiment is using a subset of the UMIST face data [33,34], face images of five different individuals, to compare the feature extraction performance of all the algorithms for later manipulation of classification. The second experiment is using a data set which contains 698 images of different poses of a face to compare the manifold learning performance of high-dimensional data set for all the algorithms.

Table 1 displays all the data sets that have been used in the experiments and briefly summaries their major characteristics, such

Please cite this article as: Y. Pan, et al., Weighted locally linear embedding for dimension reduction, Pattern Recognition (2008), doi: 10.1016/j.patcog.2008.08.024

Table 1
Experimental data sets description

Data set	Samples	Neighbor size	Dimension
Swiss roll	1000	12	3
Toroidal helix	1000	24	3
Gaussian	1000	12	3
Punctured sphere	1000	24	3
3-D clusters	1000	24	3
Different face images	100	24	10 304
Different pose images	698	8	4096

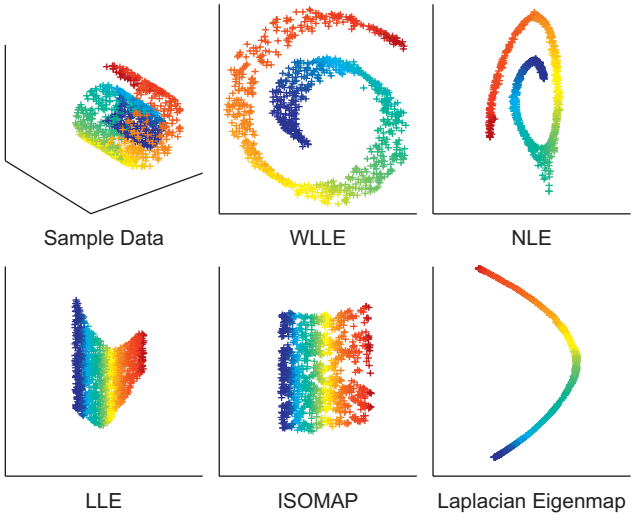

Fig. 2. Example of Swiss roll.

as the number of samples, the number of neighbors, the dimension of the data sets.

5.1. Experiments on artificial manifold

In this section, we present the experimental results of WLLE tested on several standard manifolds, and compare WLLE we proposed to other dimension reduction methods such as LLE, NLE, ISOMAP and Laplacian Eigenmaps.

For comparison of the embedding property, we have conducted all the manifolds embedding with the three algorithms, LLE, NLE and WLLE. For each data set, every algorithm is used to obtain a 2D embedding. Figs. 2–5 show the embedding results of these three algorithms for the manifold data sets. In each figure, the sampled data set is shown at the top left, in a 3D representation; the embedding result by WLLE is shown at the top middle; the result by NLE is shown at top right; the result of LLE is shown at bottom left and the result of ISOMAP and Laplacian Eigenmaps are shown at bottom middle and bottom right, respectively.

Swiss roll is a randomly sampled plane is rolled up into a spiral. Fig. 2 shows the sampled Swiss roll and the embedding results for it by the five algorithms, and we can see that the embedding effects of the five algorithms are quite different. LLE and ISOMAP unrolls the 3D data set into a plane, we can see the neighborhood relationships of LLE and ISOMAP are preserved well from the color coding, and the shape of ISOMAP's embedding is smoother than that of LLE. Both NLE and WLLE unroll the original data set to a 2D roll, while the shape of WLLE is more regular. According to the color, the neighborhood relationships in 3D are preserved in the lower dimension embeddings of these methods. The results can be viewed from different aspects. On one side, from the "manifold embedding" point of view, the goal

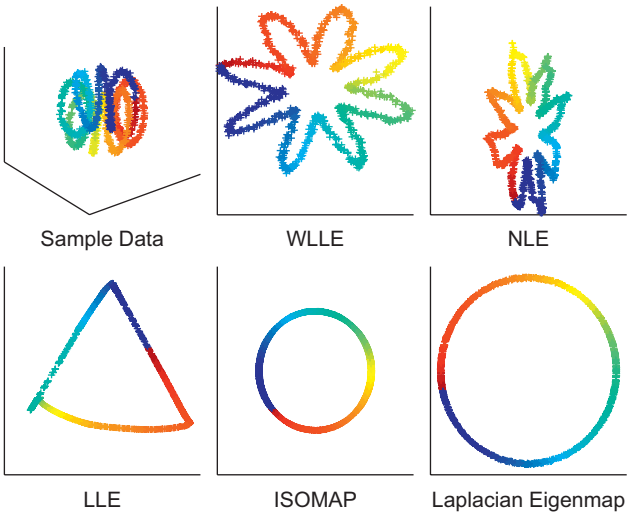

Fig. 3. Example of Toroidal helix.

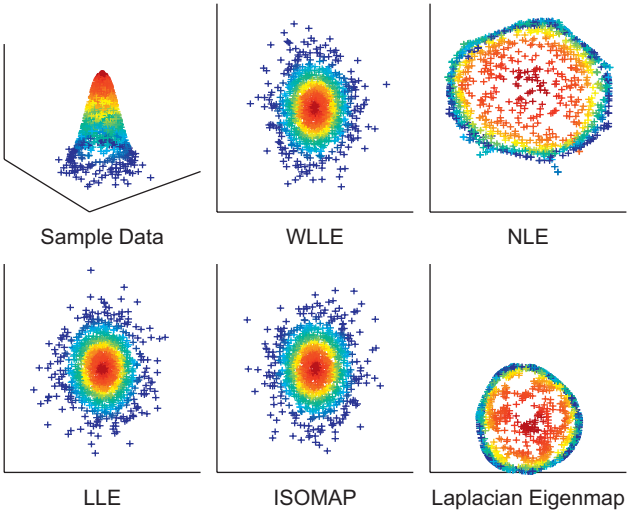

Fig. 4. Example of Gaussian distribution.

of manifold embedding is to find a Euclidean representation of the original data points, and the Isomap and LLE algorithms yield better embedding results than the NLE and WLLE. On the other side, from "feature extraction" point of view, the embedding results given by the NLE and WLLE keep both the local neighborhood relationship among the data and the global shape and distribution of the original data set, which means more intrinsic features have been reserved. The result of Laplacian Eigenmaps is an ark line which makes less sense.

Toroidal helix is a one-dimension curve coiled around a helix. There is a small amount of noise in the sampling along the helix. The dimension reduction method should unravel the coil into a circle. From Fig. 3, we can see that the LLE algorithm maps the 3D Toroidal helix into 2D circle in a shape of triangular. ISOMAP and Laplacian Eigenmaps give a 2D embedding of a perfectly regular circle, which means it uncoiled the Toroidal helix string. Both WLLE and NLE can embed the original 3D helix to a flower-like shape circle, which means more properties of the original data set are preserved in the 2D embedding. Further more, the result of WLLE has a very regular shape, while the NLE gives a deformed shape result.

Please cite this article as: Y. Pan, et al., Weighted locally linear embedding for dimension reduction, Pattern Recognition (2008), doi: 10.1016/j.patcog.2008.08.024

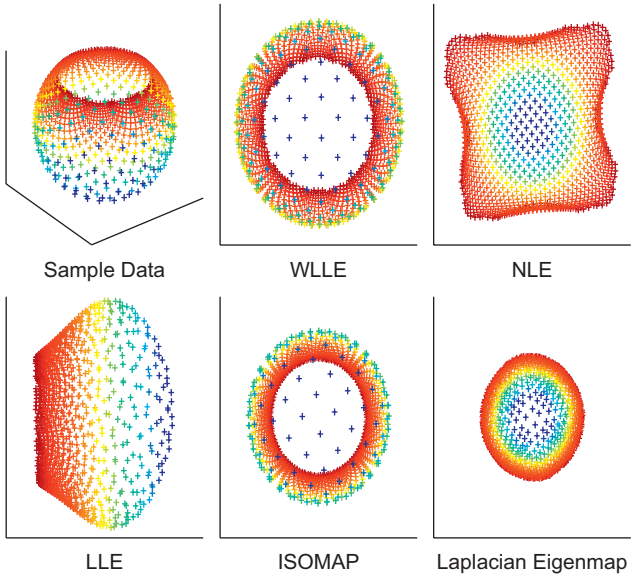

Fig. 5. Example of punched sphere.

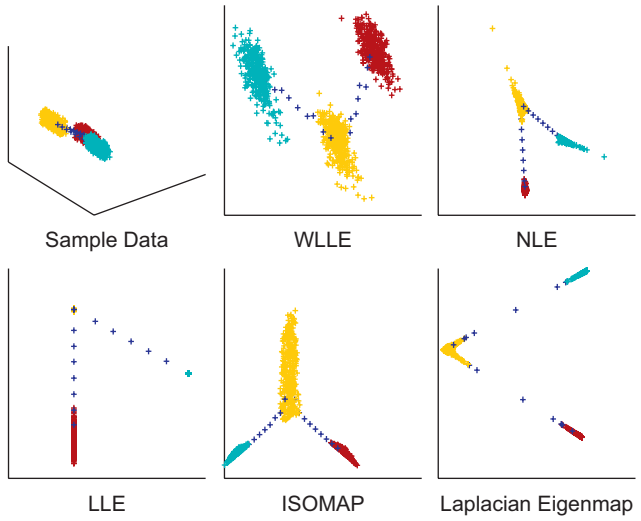

Fig. 6. 3-D clusters.

The data set of Gaussian distribution is drawn from a Gaussian distribution, a good nonlinear dimensionality reduction method should form concentric circles. In Fig. 4, from the top right and bottom right, we can see both WLLE, LLE and ISOMAP algorithm give a good concentric circle 2D embedding of the 3D Gaussian distribution, but NLE and Laplacian Eigenmaps cannot correctly embed the density property of the original data set and gives deformed shape of circles.

The punched sphere is the bottom $\frac{3}{4}$ of a sphere which is sampled no-uniformly. The sampling is densest along the top rim and sparsest on the bottom of the sphere. Its intrinsic structure should be 2D concentric circles. From Fig. 5, the LLE algorithm has some problem to correctly reconstruct the original data set in 2D embedding; the NLE algorithm gives a plane with correct distribution but the shape is not circle; the WLLE and ISOMAP algorithm give better embeddings as an exact concentric circles preserve exactly the density information of the original data set: densest along the top rim with red color and sparsest on the bottom of the sphere with blue color. The embedding given by Laplacian Eigenmaps seems the most correct 2D embedding for the 3D punched sphere. It unfold the punched sphere exactly in both density of distribution and order of colors.

5.2. Clustering

In this clusters data set, random 3D points are assigned to tight non-overlapping data clusters. Since most nonlinear dimension reduction techniques require a connected data set, the clusters are randomly connected with blue 1D line segments. A good nonlinear dimension reduction technique should preserve clusters, as shown by the color groupings. In Fig. 6, a three-clusters data set is shown at the top left, the embedding result of this data set by WLLE, NLE and LLE is shown at top right, bottom left and bottom right, respectively. We can see that the WLLE embeds the 3D clusters faithfully to 2D clusters, so does the NLE. The result of ISOMAP is also meaningful and easy to identify. However, the result of LLE is not good enough since two clusters of the three in original data set shrink to one point or even disappear in its 2D embedding, which cause problem to distinguish the clusters and the connection line between clusters. The result of Laplacian Eigenmaps may have the same problem as LLE.

Table 2
Computational time comparison

Algorithm	WLLE	NLE	LLE	ISOMAP	Laplacian Eigenmaps
Swiss roll	8.4	11.8	3.1	95.2	1.2
Toroidal helix	6.2	12.1	1.3	125.5	1.1
Gaussian distribution	7.3	10.9	3.9	100.7	0.99
Punched sphere	6.8	7.9	6.9	100.2	1.3
3D clusters	5.5	10.6	2.2	74.1	1.1

Besides the embedding performance, computational cost is also a critical issue to evaluate an algorithm. To summarize the computational cost for all these artificial manifold data sets, Table 2 shows the computational time (second) of each algorithm for all the artificial manifold data sets. From the table, WLLE has higher computational cost than LLE and Laplacian Eigenmaps, but lower computational cost than NLE and ISOMAP. All the computation are done in a computer with Pentium 4 cpu 2.8 GHz, 1 GB of ram to assure the same environment of computing.

5.3. Face images

Face recognition is the one of the most popular research topic in pattern recognition during this decade [35–37]. It can be widely used in entertainment, information security, intelligent robotics and so on. Recently, great development has been done by researchers on both algorithm and system. A critical part in face recognition is the dimension reduction algorithm for feature extraction.

In this area, global feature extraction algorithm such as PCA, LDA and all the methods based on combination of this two gave many good results in applications on facial recognition. Later, as a nonlinear extension of PCA, KPCA [9] has shown great advantages on data representation for clustering and classification of complicated facial data set. Based on the very observation that null subspace contains useful information for clustering, in Ref. [2], Lu et al. proposed KDDA, which is combination of KPCA and direct linear discriminant analysis (DLDA). Another combination of LDA and KPCA, called complete kernel Fisher discriminant (CKFD), has been proposed in Ref. [38]. All these kernel based methods have a major disadvantage in that the selection of kernel function and its parameters is usually made by trial and error or based on experience, which greatly weaken the practical value of these methods. Moreover, the final projections are related to all the training samples, so that the requirements for training samples are usually strict.

Please cite this article as: Y. Pan, et al., Weighted locally linear embedding for dimension reduction, Pattern Recognition (2008), doi: 10.1016/j.patcog.2008.08.024

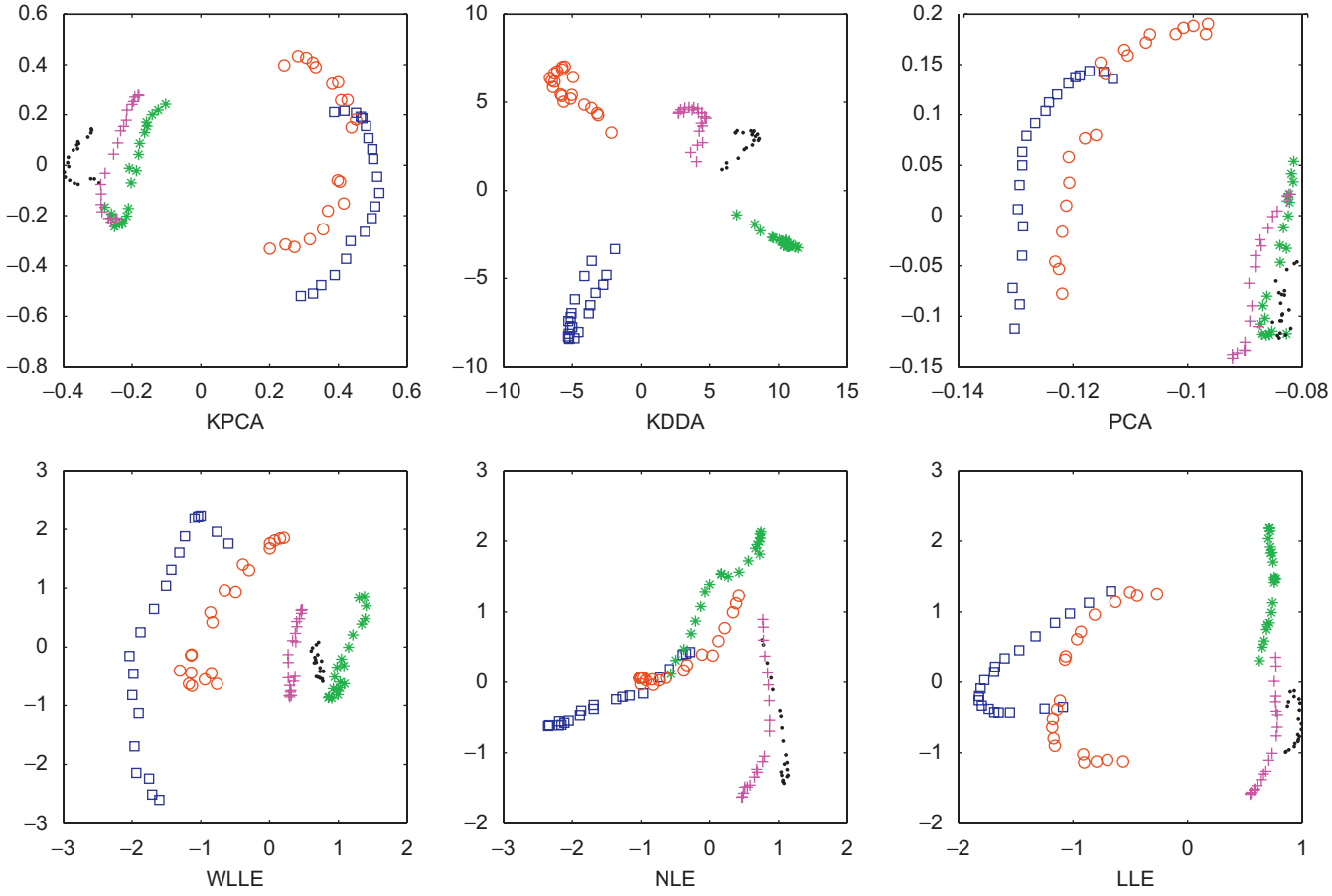

Fig. 7. Dimension reduction result of UMIST face data by six different methods.

Compare to these kernel based methods, LLE has its own advantages because of unsupervised property. On one hand, it do not need training samples, which is especially helpful for small sample size of the face pattern's distribution. On the other hand, it only has one simple parameter, K number of neighbors selected, to be chosen, which make it easy to be applied. However, the performance of original LLE will decline when the data distribution is not well or uniformed distributed. Another problem is that the algorithm is not robust to parameter changes, which will be discussed in Section 6. To combat these problems, we proposed WLLE which may have better performance for complicated data set such as face images.

5.3.1. Classification of different faces

To demonstrate the face recognition performance of WLLE and compare to other famous methods for face recognition, in this section, we utilize the UMIST face database [33,34] for experiment, which consists of 564 images of 20 people in PGM format, approximately 220 × 220 pixels in 256 shades of grey. Each covering a range of poses from profile to frontal views. Subjects cover a range of race/sex/appearance. The original 220 × 220 PGM format face images were cropped to 112 × 92 images, a standardized image size commonly used in face recognition experiment. In our experiment, we extract a typical subset of the UMIST face database, which contains face images of five different individuals, each individual has 20 face images covering range from profile to frontal views. As such, the subset we used in this experiment are 100 samples with dimensionality of 10 304 in five classes (different individuals). Six types of low-dimensional representations are produced from the face images subset by using different feature extraction algorithms, PCA, KPCA, KDDA, WLLE, LLE and NLE. For PCA, KPCA and KDDA, all of the face images in subset are used in both the training procedure to generate subspaces and the testing procedure to project them onto the generated subspaces. For each image, its projections in the first two most important features bases are visualized in the first row of Fig. 7. For WLLE, LLE, and NLE, the high-dimensional face image data are mapped into 2D embeddings, which are shown in the second row of Fig. 7.

The low-dimensional representations produced by the algorithms are quite different. Among them, the KDDA-based result and WLLE-based result showed better clustering property, but the other four algorithms result in some overlapping between different classes of the face data, which may make them non-separable. Especially, the result of KDDA is fairly linear separable, which may result from its separability criteria-based algorithm. Unlike the diffuse shape of the five classes in result by KDDA, the result by WLLE gives a parallel shape of the five classes. Although no overlapping between different classes, the short distance between clusters indicates that the WLLE-based feature representation is less linear separable than the KDDA-based result. Overall, simply inspection of Fig. 7 indicates that the feature representations produced by KDDA and WLLE outperform, in view of separability, the ones produced by PCA, KPCA, LLE and NLE. This will later proved by feeding these 2D features obtained by six algorithms into a simple SVM classifier.

Since the objective of this experiment is to compare the performance of different feature extraction algorithms, we keep the parameters of SVM unchanged during all the experiments. The performance is evaluated using average error rate of eight runs for each algorithm, which obtained by dividing total number of

Please cite this article as: Y. Pan, et al., Weighted locally linear embedding for dimension reduction, Pattern Recognition (2008), doi: 10.1016/j.patcog.2008.08.024

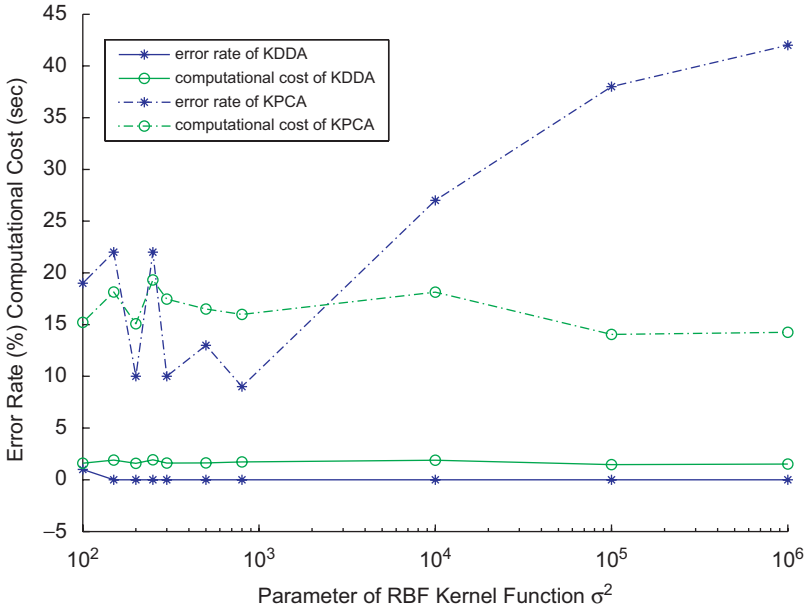

Fig. 8. Comparison of error rates and computational cost as functions of σ^2 for KPCA and KDDA.

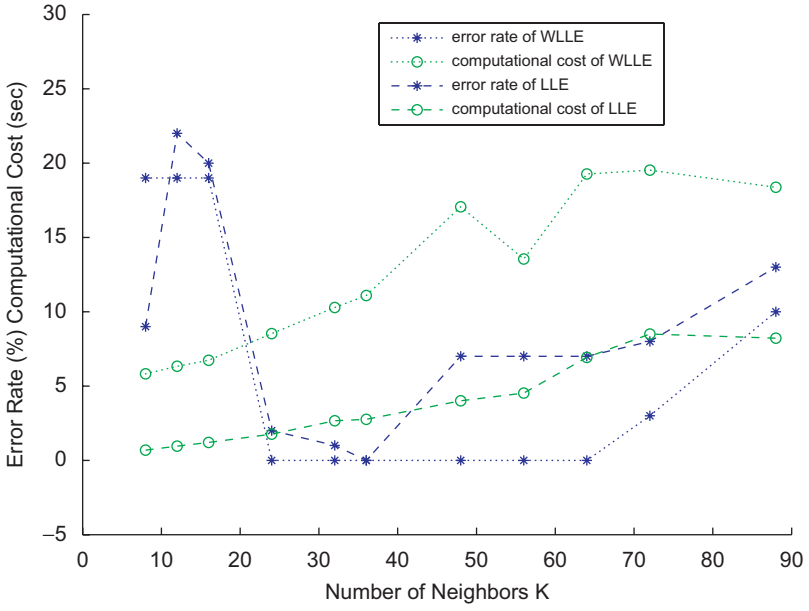

Fig. 9. Comparison of error rates and computational cost as functions of K for WLLE and LLE.

misclassifications by product of number of samples and number of runs.

Noted that the performance of kernel-based methods are greatly affected by what kernel function chosen and the parameter changes of the function, we use a RBF kernel function for both KPCA and KDDA, and record the error rate with different kernel parameter, the scale value σ^2 for RBF kernel. Fig. 8 shows the error rates and computational cost as functions of σ^2 within the rage from 1e2 to 1e7 for algorithms KPCA and KDDA. Either error rate or computational cost indicate that the KDDA algorithm outperform KPCA.

The only parameter for LLE-based methods is the number of neighbors, K. As such, we record the error rate with different K for LLE and WLLE algorithm. The NLE algorithm do not need choose parameter K. Fig. 9 shows the error rates and computational cost as functions of K within the range from 8 to 88 for algorithms WLLE and LLE. Although the computational cost of WLLE is higher than LLE, the error rate shows the classification performance of WLLE outperform that of LLE.

For comparison of the all the six algorithm, we use optimal parameter in the experiment shown in Table 3, which shows the optimal parameter ranges, average computational cost and average error rate in the optimal range. From Table 3, it can be easily observed that PCA is the most simple algorithm but the classification performance is not satisfactory. KDDA has low computational cost and excellent classification performance, and is the best algorithm for this face recognition problem. KPCA and NLE have high computational costs and average classification error rates. WLLE shows good performance for classification, but the computational cost is

Please cite this article as: Y. Pan, et al., Weighted locally linear embedding for dimension reduction, Pattern Recognition (2008), doi: 10.1016/j.patcog.2008.08.024

Table 3
Classification error rates and computational time comparison

Algorithm	Parameter	Computational time (s)	Error rate (%)
PCA	N.A.	0.5	47
KPCA	σ^2 : 2e2–8e2	16.5	10
KDDA	$\sigma^2 \geqslant$ 2e2	1.7	0
LLE	K : 24–36	2.4	1
WLLE	K : 24–72	13.3	0
NLE	N.A.	11.5	8

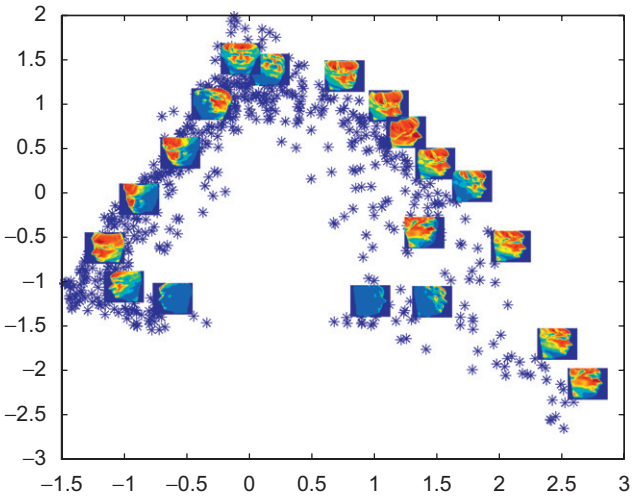

Fig. 10. 2D embeddings of different pose face images by WLLE.

relatively high compare to KDDA and LLE. Although the smallest error rates for WLLE and LLE are almost the same, one can see from Fig. 9 that WLLE gives the optimal performance for a much larger range of parameter than LLE, which means WLLE is more robust to parameter changes than LLE. This will be discussed in detail later in Sections 6.3 and 6.4.

5.3.2. Manifold learning of different poses of a face

In next simulation study, the feature extraction algorithms are used to find the coherent relationship among a set of face images [8]. This data set contains $N = 698$ gray images at a resolution of 64×64, and are different poses from left side view through front view to right side view of the same face. The input datum x_i of X is constructed by formatting the image pixel column by column from left to right and concatenation them to form the column vector.

The computed 2D embeddings by WLLE are shown in Fig. 10, several face images are shown next to the corresponding embedding point. These embeddings form an arch-bridge shape. To a certain extent, it is identical to the motion trajectory of the faces. From the left end of the arch, through middle peak till the right end of the arch, the embeddings are corresponding to the left pose face, front face and right pose face. Although the face images are high-dimensional data, the 2D embeddings of the face images are related to meaningful attributes of the motion of the subject head in the images. Thus, if a new face image is given, we can compute its corresponding embedding and identify the face direction by finding its position in Fig. 10.

For comparison, the same data are also processed by PCA, KPCA and KDDA, which are shown in Figs. 11, 12, 13, respectively. All of them show some patterns according to the different poses of face images. The PCA and KPCA map the left and right views of the face to the top and bottom part of the embedding, respectively.

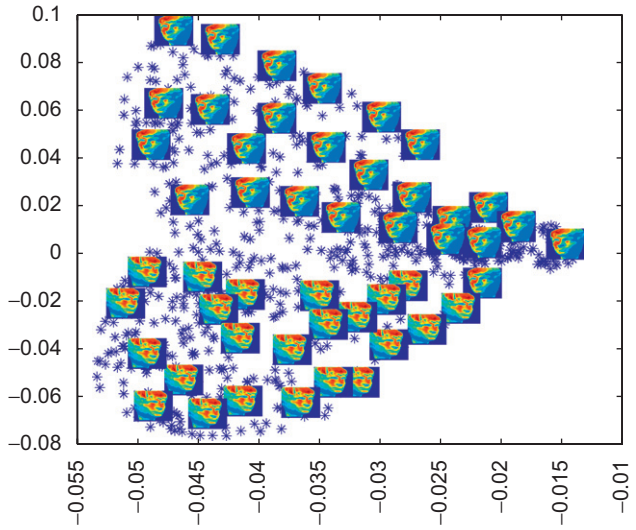

Fig. 11. 2D embeddings of different pose face images by PCA.

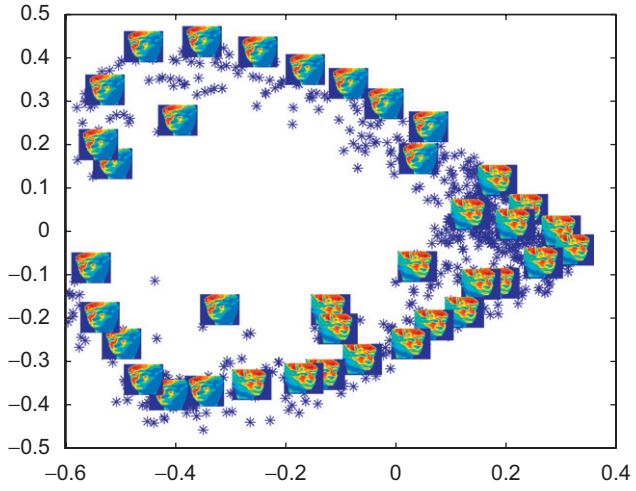

Fig. 12. 2D embeddings of different pose face images by KPCA.

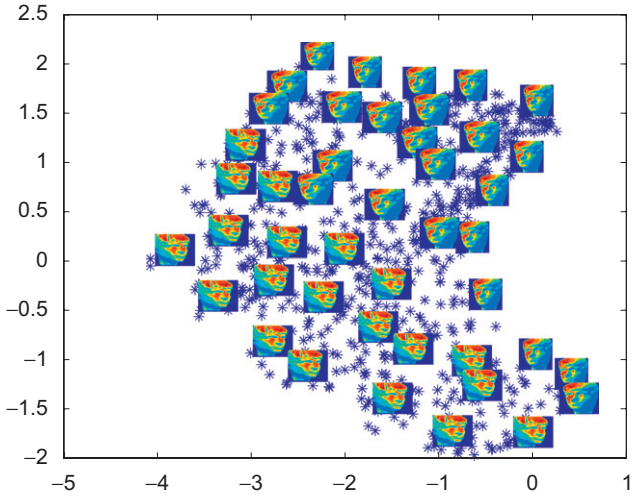

Fig. 13. 2D embeddings of different pose face images by KDDA.

Please cite this article as: Y. Pan, et al., Weighted locally linear embedding for dimension reduction, Pattern Recognition (2008), doi: 10.1016/j.patcog.2008.08.024

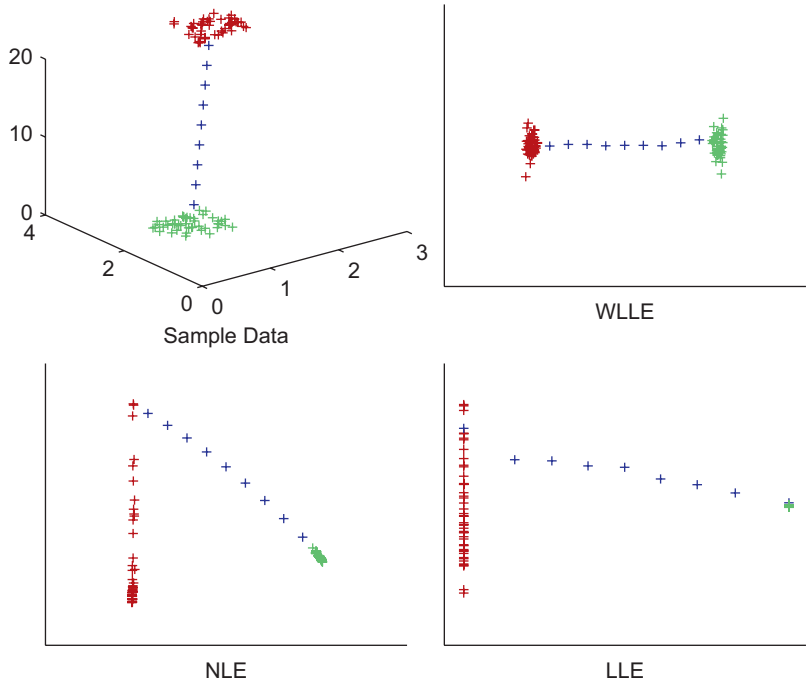

Fig. 14. A difficult data set for LLE.

KDDA maps the left and right views of the face to the top right and bottom left part of the embedding, respectively. Simple inspection of Figs. 10–13 indicates that WLLE extracts a more smooth and meaningful string of manifold for the images of different poses.

5.4. Comments to the experiment result

The WLLE method improved LLE method by using cam weighted distance measurement in neighbor selection phase to avoid redundant and insufficient neighbor selections. In this section, a number of numerical experiments have been done to fully demonstrate the properties of the proposed WLLE algorithm. Its main advantages and disadvantages can be summarized as follows.

(i) WLLE has good performance for both uniform distribution and non-uniform distribution. Especially for non-uniform distribution as in Figs. 4 and 5, WLLE has performance as good as ISOMAP, but its computational cost is much lower than ISOMAP as shown in Table 2;
(ii) WLLE is relatively robust to parameter changes as shown in Fig. 9. This property will be discussed in detail later in Sections 6.3 and 6.4;
(iii) WLLE is helpful in both human face recognition and facial poses identification as shown in Figs. 7 and 10;
(iv) compare to KDDA, WLLE cause much higher computational cost but its performance is not better, as shown in Table 3; and
(v) the performance of WLLE will decline when the number of neighbors is too small or too large, which can be seen from Fig. 9.

6. Discussion

We conclude by tracing the origin of this work, discuss the main advantages of WLLE compare to other dimension reduction methods and possible future research.

6.1. Early motivation: a difficult example for LLE

The embeddings of LLE are optimized to preserve the geometry of nearby inputs. Though the collective neighborhoods of these inputs are overlapping, the coupling between faraway inputs can be severely attenuated if the data are noisy, sparse, or weakly connected. Thus, the most common failure mode of LLE is to map faraway inputs to nearby outputs in the embedding space [5].

A difficult example for LLE was mentioned in Ref. [5], and this example is shown at the top left of Fig. 14, where the data were generated from the volume of a 3D "barbell". For this example, LLE algorithm does not lead to reasonable results. It is arguable that in this case this data set can be considered to belong to a collection of manifolds of different dimensionality. Thus, the weakly connected component cause difficult for giving faithful embedding. The possible resolution for this problem lies on identifying weakly connected components or varying the number of neighbors K per data point.

NLE uses an adaptive neighbor selection method, which determines different neighbor size for each data, to substitute the KNN or ε-neighborhood neighbor selection used in LLE. However, its simple neighbors selection algorithm used for reducing redundancy cannot help for solving the weakly connected components problem caused by multiple dimensionality. Further more, the too small neighbor size decided by NLE makes the problem even worse.

The WLLE algorithm proposed in this paper chooses neighbors based on a modified distance measurement, and this weighted distance measurement can strengthen the connection of the weakly connected components in non-uniform sampled data. Thus, based on this distance measurement, the neighborhoods selection algorithm can adopt the weakly connected components as neighbors as well as the strongly connected components, and the weakly connected components can be identified. The top right of Fig. 14 shows the embedding result by WLLE for the "barbell" data set. It is easy to see WLLE preserves the clusters and the connection line faithfully. Compare to the result of WLLE, NLE and LLE both give a worse embedding result as shown at bottom left and bottom right of Fig. 14.

Please cite this article as: Y. Pan, et al., Weighted locally linear embedding for dimension reduction, Pattern Recognition (2008), doi: 10.1016/j.patcog.2008.08.024

In the figure we can see that the clusters and connection line is difficult to be identified in 2D embeddings by NLE and LLE, and most of the properties in original 3D data set is lost.

6.2. Computational complexity analysis

All these three algorithms mainly consist of three phases: the nearest neighbor phase; the optimal reconstruction weights phase; and reconstruction of low embedding phase. Furthermore, both the NLE and the WLLE have one more steps than the LLE in nearest neighbor phase.

Given a dimension reduction problem with N data points of D dimension, the computational complexity of the nearest neighbor phase for LLE (using KNN method) is $O(DN^2)$, while the NLE has computational complexity of $O(DN^2+\alpha ND)$, with an extra step for estimation neighborhood, where α is the number of neighbors selected which is unfixed value because the NLE determines the number of neighbors adapt to the data set distribution. The WLLE has computational complexity of $O(ND+2DN^2)$ in nearest neighbor phase, contains an extra step of parameters estimation. It is noted that in this phase our new algorithm WLLE involves more computations than original LLE, but the WLLE is computational competitive with the NLE. The nearest neighbor step is simple to implement but can be time consuming for large data set ($N>10^4$). However, many techniques such as the K–D trees or ball trees can be used to compute the neighbors in $O(N\log N)$ time [1,39].

The following two steps are the same for all the three algorithms. The optimal reconstruction weights phase is typically the least expensive step of the whole algorithm, with the computation scales $O(DNK^3)$, where K is the number of neighbors decided in former step. This is the number of operations required to solve a $K\times K$ set of linear equations for each data point.

The final step, low embedding reconstruction phase, is typically the most computationally expensive, as computing the bottom eigenvectors scales as $O(dN^2)$, where d is the dimension of reconstructed embedding. However, there are many techniques currently for speeding up the eigenvector problems. For example, specialized methods for sparse, symmetric eigenvalue problems [40] can be used to reduce the complexity; for very large problems, one can consider alternative embedding cost function, such as direct descent by conjugate gradient methods [41], stochastic gradient descent [42], etc.

6.3. Stability problem

The results of LLE are typically stable over a range of neighborhood sizes. The size of that range depend on features of the data set, such as the sampling density, distribution and manifold geometry.

There are several criteria for choosing neighborhood size. First, the dimensionality of embeddings d should be strictly less than the number of neighbors, K. Some margin between d and K is generally necessary to obtain a topology-preserving embedding, but the exact relation between K and the faithfulness of the resulting embedding remains an important open problem. Second, the LLE algorithm is based on the assumption that a data point and its nearest neighbors can be modeled as locally linear; for curved data sets, choosing K too large will in general violate this assumption.

Figs. 15 and 16 shows a range of embeddings discovered by LLE algorithm and WLLE algorithm, all on the same data set but using different numbers of nearest neighbors, K. From these two figures, the results of WLLE show a wider stable range of neighbor size, but the results of LLE are easily break down as K becomes too small or large. Especially, when the number of nearest neighbors K is set too large, the embedding will jump across folds. It is difficult to faithfully unravel the manifold because that large K makes a data point and its nearest neighbors hard to be modeled as locally linear. However, the embeddings of WLLE under the same situation is much better, since the neighbor selection algorithm adopted in WLLE ensures the locally linear between data point and its neighbors even the neighbor size is quite large.

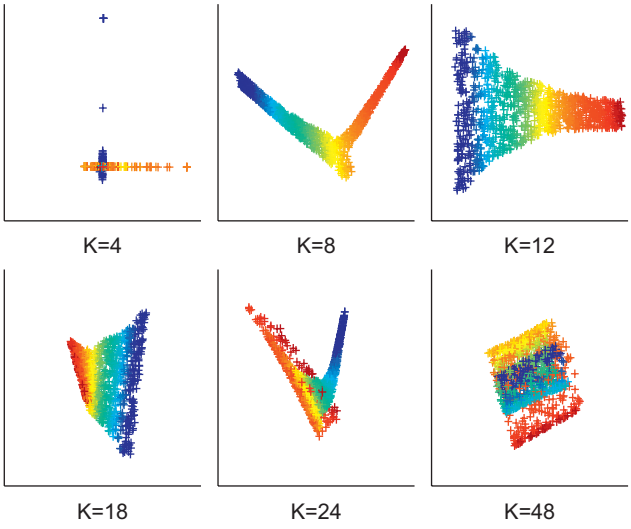

Fig. 15. Effect of neighborhood size on LLE.

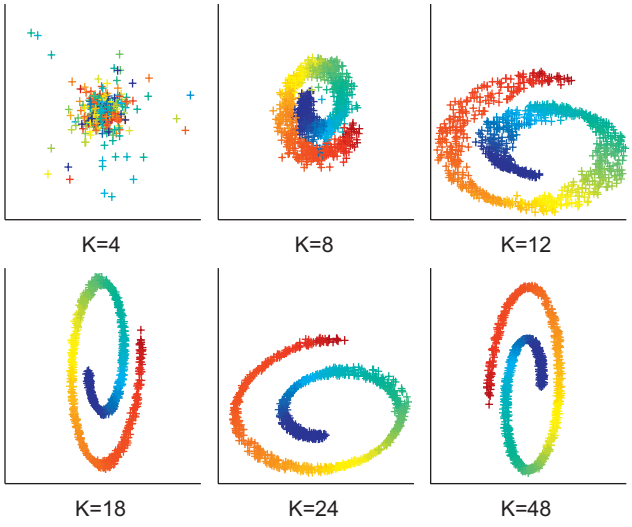

Fig. 16. Effect of neighborhood size on WLLE.

Finally, in the case that the original data is itself low-dimensional, which may result in $K>D$, the local reconstruction weights are no longer uniquely defined since each data point can be reconstructed perfectly from its neighbors. In this case, some further regularization must be added to break the degeneracy. In the procedure of calculating optimal reconstruction weights as described in Section 4, one thing should be mentioned is that in some unusual cases, the local covariance matrix (17) will be singular or nearby singular, for example when there are more neighbors than input dimensions ($K>D$), or when the data points are not in general position. In these cases, the local covariance matrix must be conditioned by adding a small multiple of identity matrix as in Eq. (21).

Please cite this article as: Y. Pan, et al., Weighted locally linear embedding for dimension reduction, Pattern Recognition (2008), doi: 10.1016/j.patcog.2008.08.024

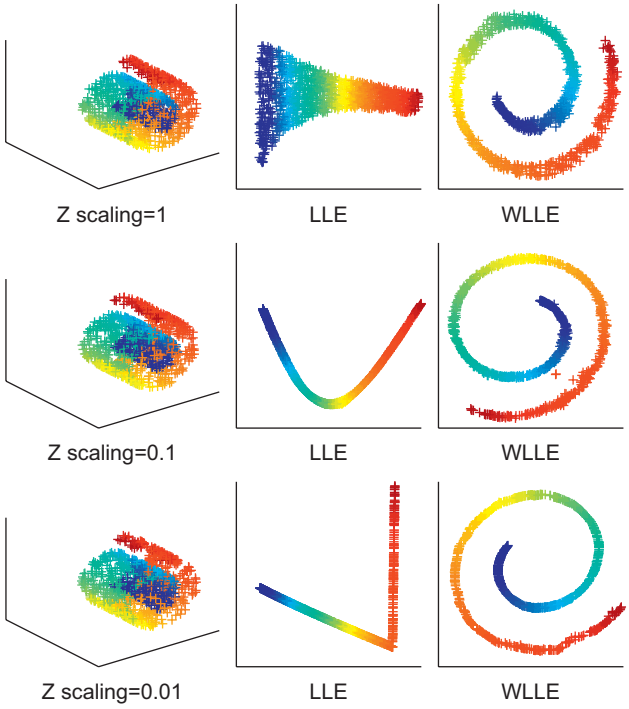

Fig. 17. Comparison of sensitivity to spiral height of Swiss roll.

6.4. Sensitivity to parameters

Besides the number of nearest neighbors K, there are many other facts affect the embedding result. Most important aspects are the features of the data set, such as the sampling density, distribution and manifold geometry. For example, Swiss roll is a randomly sampled plane which is rolled up into a spiral. One parameter of the data set is "Z scaling", the height of the spiral. It is obviously that a smaller value of "Z scaling" will make the folding more compact and thus harder to unravel. Fig. 17 shows the effect of the spiral height on embeddings of both LLE and WLLE. The left column is the original Swiss roll data with different spiral height, and the second and third column are the 2D embeddings by LLE algorithm and WLLE algorithm with respect to different spiral height, respectively.

It is shown in Fig. 17 that the spiral height of Swiss roll has great effect on the embedding results of LLE. When the "Z scaling" becomes small, the 2D representing of the Swiss roll cannot keep the regular shape well. However, the result of WLLE seems more robust, the change of spiral height has little effect on its results.

Another example of sensitivity to parameter changes is the effect of neighbor size, which is discussed in Section 6.4 and shown in Figs. 15 and 16. We know that WLLE algorithm is more robust with different neighbor size than normal LLE algorithm. For LLE, if the number of nearest neighbors K is set too large, the embedding will jump across folds.

As a result, we may conclude that the WLLE algorithm has better performance than normal LLE with parameter changes, which means the WLLE is more robust in application.

7. Conclusion

In this paper, we have presented an unsupervised learning algorithm to discover the intrinsic structures of data, such as neighborhood relationships, global distributions and clustering. The proposed algorithm optimized the process of intrinsic structure discovery by avoiding unreasonable neighbor searching, and at the same time, let the discovery adapt to the characteristics of input data set. Furthermore, it is able to discover intrinsic structures of data simultaneously, and the discovered structures can be used to compute manipulative embedding for potential classification and recognition purposes. Simulation studies and comparison with LLE and NLE demonstrated that the WLLE can give better result in manifold learning and dimension reduction and is more robust to parameter changes. Experiments on face images data sets have shown the potential of WLLE in practical problem such as face recognition.

Acknowledgments

The authors would like to thank Dr. D. Grahan and Dr. N. Allinson for providing the UMIST face database.

References

[1] S.M. Omohundro, Bumptrees for efficient function, constraint and classification learning., in: conf/nips/1990, Morgan Kaufmann, 1991, pp. 693–699.
[2] J. Lu, K.N. Plataniotis, A.N. Venetsanopoulos, Face recognition using kernel direct discriminant analysis algorithms, IEEE Trans. Neural Networks 14 (1) (2003) 117–126.
[3] P.J. Rousseeuw, A.M. Leroy, Robust Regression and Outlier Detection, Wiley, New York, 1987.
[4] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (2000) 2323–2326.
[5] L.K. Saul, S.T. Roweis, Think globally, fit locally: unsupervised learning of low dimensional manifolds, J. Mach. Learn. Res. 4 (2003) 119–155.
[6] I. Jolliffe, Principal Component Analysis, second ed., Springer, New York, 2002.
[7] T.F. Cox, M.A.A. Cox, Multidimensional Scaling, Chapman & Hall, London, 1994.
[8] J. Tenenbaum, V. de Silva, J.C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (2000) 2319–2323.
[9] B. Scholkopf, A. Smola, K.-R. Müller, Nonlinear component analysis as a kernel eigenvalue problem, Neural Comput. 10 (5) (1998) 1299–1319.
[10] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Comput. 15 (6) (2003) 1373–1396.
[11] S.S. Ge, T.H. Lee, C.J. Harris, Adaptive Neural Network Control of Robotic Manipulators, World Scientific, Singapore, 1992.
[12] X. Li, N. Roeder, Face contour extraction from front-view images, Pattern Recognition 28 (8) (1995) 1167–1179.
[13] G. Yang, T.S. Huang, Human face detection in a complex background, Pattern Recognition 27 (1) (1994) 53–63.
[14] R.D. Short, K. Fukunaga, The optimal distance measure for nearest neighbor classification, IEEE Trans. Inf. Theory 27 (5) (1981) 622–627.
[15] K. Fukunaga, L. Hostetler, Optimization of k-nearest neighbor density estimates, IEEE Trans. Inf. Theory 19 (3) (1973) 320–326.
[16] M. Muezzinoglu, J. Zurada, Rbf-based neurodynamic nearest neighbor classification in real pattern space, Pattern Recognition 39 (5) (2006) 747–760.
[17] Y. Pan, S.S. Ge, F.R. Tang, A.A. Mamun, Detection of epileptic spike-wave discharges using SVM, in: Proceedings of the 2007 IEEE International Conference on Control Applications, Suntec City, Singapore, 2007, pp. 467–472.
[18] M. Su, C. Chou, A modified version of the k-means algorithm with a distance based on cluster symmetry, IEEE Trans. Pattern Anal. Mach. Intell. 23 (6) (2001) 674–680.
[19] A. Likas, N. Vlassis, J. Verbeek, The global k-means clustering algorithm, Pattern Recognition 36 (2) (2003) 451–461.
[20] K. Fukunaga, T.E. Flick, An optimal global nearest neighbor metric, IEEE Trans. Pattern Anal. Mach. Intell. 6 (3) (1984) 314–318.
[21] J.H. Friedman, Flexible metric nearest neighbor classification, Technical Report, Department of Statistics Stanford University, Stanford, CA, USA, 1994.
[22] T. Hastie, R. Tibshirani, Discriminant adaptive nearest neighbor classification, IEEE Trans. Pattern Anal. Mach. Intell. 18 (6) (1996) 607–616.
[23] D.G. Lowe, Similarity metric learning for a variable-kernel classifier, Neural Comput. 7 (1) (1995) 72–85.
[24] C. Domeniconi, J. Peng, D. Gunopulos, Locally adaptive metric nearest-neighbor classification, IEEE Trans. Pattern Anal. Mach. Intell. 24 (9) (2002) 1281–1285.
[25] Y.G. Zhang, C.S. Zhang, D. Zhang, Distance metric learning by knowledge embedding, Pattern Recognition 37 (1) (2004) 161–163.
[26] J. Yu, J. Amores, N. Sebe, P. Radeva, Q. Tian, Distance learning for similarity estimation, IEEE Trans. Pattern Anal. Mach. Intell. 30 (3) (2008) 451–462.
[27] N. Mekuz, C. Bauckhage, J.K. Tsotsos, Face recognition with weighted locally linear embedding, in: Proceedings of the Second Canadian Conference on Computer and Robot Vision, 2005, pp. 290–296.
[28] C.Y. Zhou, Y.Q. Chen, Improving nearest neighbor classification with cam weighted distance, Pattern Recognition 39 (2006) 1–11.
[29] S.S. Ge, F. Guan, A.P. Loh, C.H. Fua, Feature representation based on intrinsic structure discovery in high dimensional space, in: Proceedings of the 2006 IEEE International Conference on Robotics and Automation, Orlando, FL, USA, 2006, pp. 3399–3404.

Please cite this article as: Y. Pan, et al., Weighted locally linear embedding for dimension reduction, Pattern Recognition (2008), doi: 10.1016/j.patcog.2008.08.024

[30] S.S. Ge, F. Guan, Y. Pan, A.P. Loh, Neighborhood linear embedding for intrinsic structure discovery, Mach. Vision Appl. J., accepted for publication.
[31] S.S. Ge, Y. Yang, T. Lee, Hand gesture recognition and tracking based on distributed locally linear embedding, in: Proceedings of IEEE International Conference on Robotics, Automation and Mechatronics, Bangkok, Thailand, 2006, pp. 567–572.
[32] E. Pekalska, A. Harol, R. Duin, B. Spillmann, H. Bunke, Non-euclidean or non-metric measures can be informative, in: Structural, Syntactic, and Statistical Pattern Recognition, 2006, pp. 871–880.
[33] D.B. Graham, N.M. Allinson, Characterizing virtual eigensignatures for general purpose face recognition, in: H. Wechsler, P.J. Phillips, V. Bruce, F. Fogelman-Soulie, T.S. Huang (Eds.), Face Recognition: From Theory to Applications, NATO ASI Series F, Computer and Systems Sciences, vol. 163, 1998, pp. 446–456.
[34] D.B. Graham, N. Allinson, URL: ⟨http://images.ee.umist.ac.uk/danny/database.html⟩, 1998.
[35] C.S. Zhang, J. Wang, N.Y. Zhao, D. Zhang, Reconstruction and analysis of multi-pose face images based on nonlinear dimensionality reduction, Pattern Recognition 37 (2) (2004) 325–336.
[36] C.H. LEE, J.S. Kim, K.H. Park, Automatic human face location in a complex background using motion and color information, Pattern Recognition 29 (11) (1996) 1877–1889.
[37] Y. Dai, Y. Nakano, Face-texture model based on sgld and its application in face detection in a color scene, Pattern Recognition 29 (6) (1996) 1007–1017.
[38] J. Yang, A.F. Frangi, J.yu. Yang Zhong Jin, KPCA plus LDA: a complete kernel fisher discriminant framework for feature extraction and recognition, IEEE Trans. Pattern Anal. Mach. Intell. 27 (2) (2005) 230–244.
[39] A.G. Gray, A.W. Moore, N-Body problems in statistical learning, in: conf/nips/2000, 2001, pp. 521–527.
[40] D.R. Fokkema, G.L.G. Speijpen, H.A. Vandervorst, Jacobi-Davidson style QR and QZ algorithms for the reduction of matrix pencils, SIAM J. Sci. Comput. 20 (1) (1998) 94–125.
[41] W. Press, S.A. Teukolsky, W.T. Vetterling, B. Flannery, Numerical Recipes in C: The Art of Scientific Computing, second ed., Cambridge University Press, New York, NY, USA, 1993.
[42] Y. LeCun, G.B. Orr, K.-R. Müller, Efficient backprop, in: G.B. Orr, K.-R. Muller (Eds.), Neural Networks: Tricks of the Trade, Springer, Berlin, 1998.

About the Author—YAOZHANG PAN received the B.Eng. and the M.Eng. degrees from Harbin Institute of Technology, China in 2004 and 2006, respectively. She is currently working toward a Ph.D. degree in the Department of Electrical and Computer Engineering at the National University of Singapore. His research interests include machine learning, dimension reduction, pattern recognition, EEG signal processing, social robotics and brain robot interface.

About the Author—SHUZHI SAM GE, IEEE Fellow, Ph.D., DIC, B.Sc., P.Eng., is founding Director of Social Robotics Lab of Interactive Digital Media Institute, and Director of Edutainment Robotics Lab of the Department of Electrical and Computer Engineering, the National University of Singapore.
He has (co)-authored three books: Adaptive Neural Network Control of Robotic Manipulators (World Scientific, 1998), Stable Adaptive Neural Network Control (Kluwer, 2001) and Switched Linear Systems: Control and Design (Springer, 2005), and over 300 international journal and conference papers. He is a co-founder of Personal E-Motion Pte Ltd dedicated to interactive multimedia digital books for education and digital publishing. He is the founding Editor-in-Chief, International Journal of Social Robotics, Springer. He has served/been serving as an Associate Editor for a number of flagship journals including IEEE Transactions on Automatic Control, IEEE Transactions on Control Systems Technology, IEEE Transactions on Neural Networks, and Automatica. He also serves as a book Editor of the Taylor & Francis Automation and Control Engineering Series. His current research interests include social robotics, multimedia fusion, adaptive control, intelligent systems and artificial intelligence.

About the Author—ABDULLAH AL MAMUN received B.Tech. (Hons.) degree from the Indian Institute of Technology, Kharagpur, India in 1985, and the Ph.D. degree from the National University of Singapore in 1997.
In his professional career, he worked as Research Engineer at the Data Storage Institute, Singapore and as Staff Engineer at Maxtor Peripherals prior to joining the faculty of the department of Electrical and Computer Engineering, National University of Singapore. His research interest includes precision servomechanism, mechatronics, intelligent control, and autonomous mobile robots.

Please cite this article as: Y. Pan, et al., Weighted locally linear embedding for dimension reduction, Pattern Recognition (2008), doi: 10.1016/j.patcog.2008.08.024