Robust local tangent space alignment via iterative weighted PCA

Yubin Zhan,*1 Jianping Yin

Computer School, National University of Defense Technology, Changsha, Hunan 410073, China

Article Info

Available online 20 February 2011

Keywords:
Manifold learning
Weighted PCA
Local tangent space alignment
Robust statistics

Abstract

Recently manifold learning has attracted extensive interest in machine learning and related communities. This paper investigates the noise manifold learning problem, which is a key issue in applying manifold learning algorithm to practical problems. We propose a robust version of LTSA algorithm called RLTSA. The proposed RLTSA algorithm makes LTSA more robust from three aspects: firstly robust PCA algorithm based on iterative weighted PCA is employed instead of the standard SVD to reduce the influence of noise on local tangent space coordinates; secondly RLTSA chooses neighborhoods that are well approximated by the local coordinates to align with the global coordinates; thirdly in the alignment step, the influence of noise on embedding result is further reduced by endowing clean data points and noise data points with different weights into the local alignment errors. Experiments on both synthetic data sets and real data sets demonstrate the effectiveness of our RLTSA when dealing with noise manifold.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

Manifold learning has widespread applications in pattern recognition, computer vision and machine learning. Since the seminal work of Tenenbaum [1] and Roweis [2,3], which propose Isomap and Locally Linear Embedding (LLE) respectively, manifold learning has attracted extensive research interests from machine learning and computer vision communities. Many novel and effective manifold learning algorithms such as Hessian LLE [4], Laplacian Eigenmap [5], Riemannian Manifold Learning [6] and Local Tangent Space Alignment (LTSA) [7] have been developed recently. All these algorithms assume that the sampled high-dimensional data lie on a low-dimensional manifold, and they attempt to recover the low-dimensional manifold structure. They can all obtain expected results when this basic assumption for sampled data is true. However, in real world applications the sampled data always have noise and outliers, and the developed algorithms are sensitive to them (hereafter, we regard all data points that are not confined precisely to the smooth manifold as noise, and we will not distinguish outliers from noise). Therefore, designing robust manifold learning algorithm plays an important role in applying manifold learning to practical problems.

Owing to the importance of robust manifold learning, there exist some extensions of existing manifold algorithms to deal with the noise manifold learning problem [8–12]. Most of these existing extensions work as a preprocess to detect noise and to reduce their influence before manifold learning algorithm performs. For example, Chang et al. address the noise problem in the context of LLE [8]. They adopt robust Principal Component Analysis (PCA) technique to estimate the reliability scores of each point, then weighted reconstruct error with these reliability scores as weights is utilized to recover the low-dimensional embedding. The method proposed in [9] detects noisy data points and simply discards them before the embedding process.

To our best knowledge, however, there is no extension of LTSA to address the noise problem. Although the modified Local Tangent Space Alignment (MLTSA) [13] proposed by Wang has the ability of dealing with noise on some local patches, since it mainly addresses LTSA’s failure mode on large curvature manifold, it has no such ability to deal with noise distributed on the whole manifold. This paper investigates noise manifold learning problem in the context of LTSA and proposes a robust version of LTSA algorithm called RLTSA based on elaborated robust PCA algorithm.

In the following, we first review the original LTSA algorithm, then the robust PCA algorithm is introduced in Section 3; Section 4 gives details of the proposed RLTSA algorithm; experimental results are given in Section 5, followed by a conclusion in Section 6.

2. LTSA and its sensitivity to noise

The basic idea of LTSA is that the neighborhood of each point can be approximately represented by local tangent space coordinates. Details and derivation of LTSA can be found in Ref. [7].
Given a data set \( X = \{x_1, x_2, \ldots, x_n\} \) sampled from a \( d \)-dimensional manifold \( \mathcal{M} \) where column vector \( x_i \in \mathbb{R}^d (D > d) \) represents a sample, the main steps of LTSA are as follows:

1. **(Extracting local information)** For each sample \( x_i \) \( (i = 1, 2, \ldots, n) \):
   - (a) Construct neighborhood \( X_i = \{x_i, x_{i_1}, \ldots, x_{i_k}\} \) (here we use \( k \) nearest neighbors including itself in terms of Euclidian distance in input space).
   - (b) By SVD of centered matrix \( X_i - \bar{X}_i e_i^T \) where \( \bar{X}_i = (1/k) \sum_{j=1}^{k} x_j \) and \( e \) is a column vector of all 1’s with suitable dimensionality, the orthogonal basis \( U_i \) of \( d \)-dimensional tangent space consists of the \( d \) left singular vectors of matrix \( X_i - X_i e_i^T \) corresponding to the first \( d \) largest singular values. Then the local tangent space coordinates of neighborhood \( X_i \) can be computed as \( \hat{X}_i = [\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n] = U_i^T (X_i - \bar{X}_i e_i^T) \).

2. **(Constructing alignment matrix)** Denote the \( d \)-dimensional embedding coordinates by \( Y = \{y_1, y_2, \ldots, y_n\}, Y_i = \{y_1, y_2, \ldots, y_{i_k}\} \), which consists of the subset of column \( Y \), and are the corresponding \( d \)-dimensional embedding coordinates of neighborhood \( X_i \). The local alignment matrix can be computed by minimizing the following local alignment error:

   \[
   E_i = \min_{\hat{X}_i} \sum_{i=1}^{n} \|y_{ij} - (c_i + L \hat{x}_i)\|^2
   \]

   \[
   E_i = \min_{\hat{X}_i} \|Y_i - (c_i e_i^T + L \hat{X}_i)\|^2
   \]

   \[
   E_i = \|Y_i - L \hat{X}_i\|^2 = \|Y_i S_i \|^2
   \]

   where \( S_i \) is the 0–1 selection matrix such that \( Y_i = YS_i \) and \( \| \cdot \|^2 \) is the matrix Frobenius norm, we call \( \Phi_i \) the local alignment matrix. Then the global alignment matrix \( \Phi \) can be computed by the following formula:

   \[
   \Phi = \sum_{i=1}^{n} S_i \Phi_i
   \]

3. **(Obtaining global coordinates)** The embedding coordinates \( Y \) can be achieved by minimizing the local alignment error sum of all neighborhoods:

   \[
   \min_{\Phi} \sum_{i=1}^{n} E_i = \min_{\Phi} \sum_{i=1}^{n} \|Y_i S_i \|^2
   \]

   \[
   \min_{\Phi} \sum_{i=1}^{n} E_i = \min_{\Phi} \left( \text{tr} \left( \sum_{i=1}^{n} Y_i S_i \Phi \right)^2 \right)
   \]

   \[
   \min_{\Phi} \sum_{i=1}^{n} E_i = \min_{\Phi} \text{tr} (Y \Phi F^T Y^T)
   \]

   then \( Y = [v_1, v_2, \ldots, v_{d+1}]^T \), where \( v_i \) is the eigenvector of matrix \( \Phi F \) corresponding to \( i \)th smallest eigenvalues.

There are three issues that should be emphasized in the above original LTSA algorithm.

The first one is that LTSA is sensitive to noise. We use the following example to illustrate its sensitivity to noise.

**Example.** Considering 1500 examples sampled from the 2-D Swiss Roll manifold, we select 150 examples at random, and impose uniformly distributed noise to them. Fig. 1(a) plots these data points, where the black points are points with noise, and the colored points are clean data points. As can be seen from Fig. 1(b), due to noise, LTSA algorithm cannot recover the manifold structure well.

---

**Fig. 1.** Results of four algorithms on noise corrupted data set sampled from Swill Roll manifold. (a) Data sets; (b) LTSA result; (c) MLTSA result; (d) RLLE result; (e) RLTSA result. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
From the main steps of LTSA, one can see that recovering the real local tangent space is the key issue that relates to whether LTSA can discover the true manifold structure faithfully. In the presence of noise, however, the recovered tangent space in step 1(b) of LTSA will deviate from the real one because of the sensitivity of standard SVD technique to noise, then it will further influence the embedding result of LTSA. On the one hand, noise points will result in neighborhoods whose intrinsic dimensionality is larger than d, then d-dimensional local coordinates derived by standard SVD technique cannot catch the local geometric well; on the other hand, the orthogonal basis $U_i (i = 1, 2, \ldots, d)$ computed by standard SVD technique will contain undesired information, then the recovered tangent space will deviate from the real ones.

Note that the left singular vector of $X_i - \mathbf{x}_i e d^T$ corresponding to the j-th largest singular value is the eigenvector $u_j^{(0)}$ of covariance matrix $(X_i - \mathbf{x}_i e d^T)(X_i - \mathbf{x}_i e d^T)^T$ corresponding to the j-th largest eigenvalue; thus the orthogonal basis of tangent space can be represented as $U_i = [u_j^{(0)}, u_j^{(2)}, \ldots, u_j^{(d)}]$, and one can conclude that LTSA algorithm in fact uses the leading d-dimensional principal subspace to approximate the tangent space at each neighborhood. Therefore, robust PCA algorithm, which can find the robust principal subspace in the presence of noise, can reduce the influence of noise on recovering the local tangent space.

The second issue we emphasize is that LTSA gives each point the same weight when minimizing the local alignment error in Eq. (1). Obviously, in noise case, we should align the clean data points more accurately with the global coordinates than noise data points. This can be implemented by minimizing the local weighted alignment error instead of errors in Eq. (1).

The third issue is that there is no need to minimize the local alignment error sum of all neighborhoods [14–16]. If local tangent space coordinates derived from each neighborhood can characterize the local geometric well, they are heavily redundant. One can still obtain the embedding with marginal changes by choosing a subset of neighborhoods and minimizing local alignment error sum of these selected neighborhoods as long as the subset of selected neighborhoods satisfies the following two preconditions: (1) the selected neighborhoods together can cover all data points; and (2) each selected neighborhood can overlap with others under a certain overlapping factor such that the local tangent coordinates can be aligned to global coordinates [14]. In addition, in noise case, forcibly aligning the local coordinates of neighborhoods that are not well approximated will result in fatal error in the final embedding. Thus in our RLTSa algorithm, we select neighborhoods that are approximated well by the local tangent space coordinates, and minimize the alignment error sum of these selected neighborhoods.

According to the above analysis of LTSA algorithm, our RLTSa algorithm will make LTSA more robust from three aspects: (1) robust PCA algorithm is used instead of standard SVD to reduce the effect of noise on local tangent space coordinates so that the local coordinates reflect more accurately the local geometric of the manifold; (2) RLTSa selects neighborhoods that are well approximated by the local tangent space coordinates to align with the global coordinates; (3) in the alignment step, the influence of noise on embedding result is further reduced by-endowing clean data points and noise data points with different weights into the alignment errors.

In the next section, we will introduce the robust PCA algorithm used in our RLTSa algorithm.

### 3. Robust PCA based on iterative weighted PCA

In step 1(a) of the LTSA algorithm, the local tangent coordinates are obtained by performing PCA on each neighborhoods. However, due to the sensitivity to noise of classical PCA, the original LTSA is sensitive to noise. Many methods have been proposed to make PCA nonsensitive to noise. Some methods replace the classical covariance with the robust covariance estimator such as Minimum Covariance Determinant (MCD) estimator [17]; and some use projection pursuit techniques [18,19]. Here, we introduce a simple and effective robust PCA based on iterative weighted PCA to replace the standard PCA in step 1(a) of LTSA. The introduced robust PCA can obtain robust local tangent space coordinates, moreover, it can give us a measure on how likely each data point comes from the underlying data manifold.

#### 3.1. Classical Principal Component Analysis

PCA is a classical subspace method and has widespread applications in pattern recognition and computer vision [20]. For high dimensional data matrix $X = [x_1, x_2, \ldots, x_n] \in \mathbb{R}^{D \times n}$, the goal of classical PCA is to project $X$ into a d-dimensional ($d < D$) linear subspace such that the projected data points are as close as possible to the original data points. This can be formalized as follows:

$$
\min_{U} \sum_{j=1}^{n} \| x_j - \mathbf{U} \mathbf{U}^T (x_j - \mathbf{x}) \|^2
$$

where $U$ is a $D \times d$ projection matrix, $U^T U = I$ and $\mathbf{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$ is the mean vector. The optimal projection matrix $U$ is given by $U = [u_1, u_2, \ldots, u_d]$, where $u_i$ is the eigenvector of covariance matrix $(X - \mathbf{x} e d^T)(X - \mathbf{x} e d^T)^T$ corresponding to the ith largest eigenvalue.

#### 3.2. Robust PCA

From Eq. (8), one can see that the classical PCA obtains the optimal projection matrix $U$ by minimizing the projection error sum of all points. So it is sensitive to noise. A natural and simple way to make it more robust is to minimize the weighted projection error sum which gives each point a weight according to its projection error. Hence the optimization objective for robust PCA can be expressed as

$$
\min_{U} \sum_{j=1}^{n} w_j \| x_j - \mathbf{U} \mathbf{U}^T (x_j - \mathbf{x}) \|^2
$$

where the weights $w_j (j = 1, 2, \ldots, n)$ are determined by the projection error $e_j = \frac{1}{n} \sum_{i=1}^{n} x_i - \mathbf{w}_j / \sum_{j=1}^{n} w_j$.

For fixed weights $w_j (j = 1, 2, \ldots, n)$, the minimization problem (11) is called weighted PCA. To make it more robust, following the ideas from robust statistics [21], we adopt the following weight function:

$$
w_j = \begin{cases} 
1, & e_j \leq 1/2\tau \\
(\tau/2e_j), & e_j > 1/2\tau
\end{cases}
$$

where $\tau = \frac{1}{n} \sum_{i=1}^{n} e_i$ is the mean value of projection error.

Then the weight vector $w = (w_1, w_2, \ldots, w_n)$ depends on $e_j (j = 1, 2, \ldots, n)$ which depends on projection matrix $U$. However, $U$ in turn depends on weight vector $w$. Because of this cyclic dependency, we exploit an iterative procedure to obtain the solution. The details of the iterative procedure are described in Fig. 2.
4. RLTSa algorithm

Our RLTSa algorithm first performs robust PCA on each neighborhood to obtain local coordinates, then it aligns the local coordinates with the global coordinates via minimizing the local weighted alignment error. Finally, instead of minimizing the local alignment error sum of all neighborhoods, RLTSa computes global coordinates via minimizing local alignment error sum of elaborately selected neighborhoods.

For convenience of presentation, we first give an overview of the proposed RLTSa algorithm in Section 4.1, then some details will be interpreted in the following subsections.

4.1. RLTSa algorithm

The algorithmic procedure of RLTSa can be formulated as follows:

Step1: Construct neighborhood of each data point as the original LTSA algorithm does.

Step2:

(a) Perform robust PCA on each neighborhood to obtain the local coordinates.
(b) Determine the alignment weight $W_i$ for point $x_i$ ($i=1,2,...,n$).
(c) Determine regular neighborhoods subset $RN$.

Step3: Align local coordinates with the global embedding coordinates. For each neighborhood $X_i \in RN$, the local alignment matrix $\Phi_i$ is obtained via minimizing the following local weighted alignment error instead of local alignment error in (1):

$$E_i = \min_{\phi_i \in S_o, \phi_i \in X_i} \sum_{j=1}^{n} W_j \| y_j - (c_i + L_i \phi_i) \|^2$$

It can be rewritten in the following matrix form:

$$E_i = \min_{\phi_i \in S_o, \phi_i \in X_i} \| (Y - (c_i I + L_i \Phi_i))W \|_F^2$$

where $W = \text{diag}(\sqrt{W_1}, \sqrt{W_2}, ..., \sqrt{W_n})$. Then we substitute the optimal solution into Eq. (14), and the minimal local weighted alignment error can be expressed as follows:

$$E_i = \| Y \Phi_i \|_F^2$$

Step4: To obtain global coordinates, our RLTSa minimizes local weighted alignment error sum of neighborhoods in $RN$:

$$\min_{X_i \in RN} \sum_{i=1}^{n} E_i = \min_{X_i \in RN} \sum_{i=1}^{n} \| Y \Phi_i \|_F^2$$

The solution can be easily obtained by the eigen decomposition of the matrix $\Phi = \sum_{X_i \in RN} \Phi_i \Phi_i^T$.

Details of the key step 1(b)(c) of our RLTSa algorithm will be discussed below.

4.2. Local weighted alignment error

In LTSA algorithm, when finding the local alignment matrix $\Phi_i$, all points are given the same alignment error weight in (1). This may make the produced embedding results sensitive to noise. Therefore, our RLTSa algorithm will minimize the weighted alignment error sum in (13):

$$\sum_{i=1}^{n} W_i = \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij}$$

A natural problem is how to determine the weight for each point. The robust PCA provides us an effective tool. For neighborhood $X_i = [x_i, x_j, ..., x_n]$, after performing local robust PCA on it, each point $x_i$ in it has an associated weight $w_i$. A normalized weight can be computed as $w_i = w_i / \sum_{i=1}^{n} w_i$. For other points that are not in $X_i$, their normalized weights are set to 0. Denote point $x_i$ ($j=1,2,...,n$)'s normalized weight in $X_i$ by $W_{ij}$. This normalized weight gives us a measure on how likely $x_i$ comes from the underlying data manifold that we can infer from neighborhood $X_i$. Then after performing robust PCA on all neighborhoods, we can obtain the total likelihood of each point by summing up the normalized weight values from all neighborhoods. Then the alignment weight $W_i$ in Eq. (13) is set to this total likelihood. It can be formally expressed as follows:

$$W_i = \sum_{j=1}^{n} W_{ij}$$

One property of the alignment weight is that the mean weight over all points is equal to one, i.e.:

$$W = \frac{\sum_{i=1}^{n} W_i}{n} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} W_{ij}}{n} = 1$$

For noise point, the weight value is much small, while the weight value of clean data points is relatively large.

4.3. Choosing neighborhoods to align

To further reduce the influence of noise, RLTSa selects some neighborhoods to align with the global coordinates. In most situations robust principal subspace of each neighborhood obtained by robust PCA can approximate the real tangent space well. However, for some neighborhoods with significant influence of noise data points, their robust principal subspace cannot well approximate the real local tangent space of manifold.

Here we qualitatively classify neighborhoods into three classes:

1. Regular neighborhood: In each neighborhood of this type, clean data points from the same patch on the manifold are preponderant, so that robust principal subspace can well approximate the real manifold tangent space on that neighborhood.
2. Noise neighborhood: Conversely, in this type of neighborhood noise data points are preponderant, so the robust principal subspace cannot well approximate the real tangent.
3. “Short circuit” neighborhood: This type of neighborhood contains data points from different patches of manifold due to connection effect of noise points.

An illustration of these three types of neighborhoods is shown in Fig. 3. Obviously, robust principal subspace of regular neighborhoods can approximate the local tangent space better than...
For a regular neighborhood, since it contains much more clean data points, its reliability score should be large, in contrast, for noise neighborhood and "short circuit" neighborhood, they contain much more noise points, hence their reliability scores are much small. Following the ideas of robust statistics [21], we set a cutoff value as much small. Following the ideas of robust statistics [21], we set a cutoff value as $\frac{1}{100}$.

For a regular neighborhood, since it contains much more clean data points, its reliability score should be large, in contrast, for noise neighborhood and "short circuit" neighborhood, they contain much more noise points, hence their reliability scores are much small. Following the ideas of robust statistics [21], we set a cutoff value as much small. Following the ideas of robust statistics [21], we set a cutoff value as $\frac{1}{100}$.

Hence, RLTSA algorithm selects the regular neighborhoods to align with the global coordinates to obtain the final embedding. For this purpose, we define a reliability score $s_j$ for neighborhood $X_i$ as follows:

$$s_j = \sum_{X_i \subset X_j} W_j$$

(20)

For a regular neighborhood, since it contains much more clean data points, its reliability score should be large, in contrast, for noise neighborhood and "short circuit" neighborhood, they contain much more noise points, hence their reliability scores are much small. Following the ideas of robust statistics [21], we set the cutoff value as $c = \frac{\sum_{i=1}^{n} s_i}{(2n)}$, then we construct the regular neighborhood subset as:

$$RN = \{X_i : s_i \geq c\}$$

4.4. Variants of RLTSA and comparing with RLLE

The above RLTSA is only one simple and effective implementa-
tion of our RLTSA algorithm. For different types of neighborhoods, we can devise a sophisticated mechanism to give them different weights instead of discarding the neighborhoods that are not well approximated by local coordinates. Although discarding them will not change the embedding result, it causes that a few points cannot be covered by the selected neighborhoods and violates the integrity of data.

Both RLTSA and RLLE adopt robust PCA technique and the weighted cost function to reduce influence of noise, however, there are some significant differences between them.

Firstly, RLTSA replaces the standard SVD in LTSA with robust PCA based on iterative weighted PCA, denoise procedure is seamlessly integrated into the learning algorithm. The computation of robust local coordinates of each neighborhood and obtaining the weight of each point are in the same procedure, whereas RLLE adds the robust PCA as a preprocess into the LLE algorithm. What is more, the optimization objective of robust PCA is different from that of RLLE in the step of extraction local neighborhood relationship, which uses the result of robust PCA. This inconsistency has significant influence on the performance of RLLE when dealing with large amount of noise.

Secondly, unlike RLLE’s reconstruction of each neighborhood with clean data points to reduce influence of noise, our RLTSA does not need to do so, and it adopts weighted alignment error in each neighborhood to reduce the influence of noise.

Furthermore, RLTSA distinguishes neighborhoods according to the importance of all points in it, and has a corresponding mechanism to further reduce the influence of neighborhoods that cannot be fitted well by local coordinates. Whereas in RLLE, the weight of neighborhood of each point is given by the reliability score of that point, this is likely not so suitable.

5. Experimental results

In this section we will give the experimental results of our RLTSA algorithm on both synthetic data sets and real world data sets.

5.1. Synthetic data

We apply our RLTSA to data sets sampled from Swiss Roll manifold and helix curve, respectively. First we show the effectiveness of our RLTSA algorithm on noise manifold. We compare four algorithms LTSA [7], RLLE [8], MLTSA [13] and our RLTSA. The results are shown in Figs. 1 and 4. For data set sampled from Swiss Roll manifold which are embedding from $R^2$ to $R^2$, the performance of algorithms can be seen by taking into account the coloring of the data points in the plots. From Fig. 1, we can see that only the embedding results obtained by RLLE and our RLTSA vary the color smoothly, this means that only RLLE and RLTSA algorithm can recover the manifold structure well.

For data set sampled from helix curve, the performance can be seen from the smoothness of the curve where the x-axis is the 1-D embedding coordinates obtained by algorithm and the y-axis is the generating parameter of helix curve. The results in Fig. 4 show that only our RLTSA algorithm results in a straight line, which means that our RLTSA can accurately recover the generating parameter even in the presence of noise. Here we should give special attention to the result of MLTSA. In MLTSA algorithm, it introduces a new parameter $c$ to determine the dimensionality of local coordinates of each neighborhood. If we let the algorithm automatically choose value for $c$, in the presence of noise on the whole manifold, MLTSA algorithm will use 2-D local coordinates to represent most of the neighborhoods and then further cause that the final 1-D embedding coordinates have little relation to the generating parameter. Thus we can see that the plotted points in Fig. 4(b) form a 2-D surface rather than a curve. If we manually specify a value for $c$ such that MLTSA uses 1-D local coordinates to represent each neighborhood, then MLTSA will produce the same result as LTSA.

From the above experiments we can conclude that the RLTSA algorithm can significantly improve the performance of the LTSA algorithm on noise corrupted data.

To evaluate the performance of RLTSA algorithm on different types of noise, we perform more experiments on the data set sampled from Swill Roll manifold. This time we use Gaussian noise instead of the uniformly distributed noise. Meanwhile, we further study how the performance of RLTSA varies as the level of noise increases. Fig. 5 shows the results of RLLE and RLTSA on data set with different levels of Gaussian noise, respectively. We can see that when there are more than 20% noise points, RLLE cannot recover the manifold structure. However, for our RLTSA, even if there are 50% noise points, it can still recover the
manifold structure well. As the analysis in Section 4.4, the poor performance of RLLE on large amount of noise may be caused by the inconsistency of the optimization objective between robust PCA and the extraction local neighborhood relationship step of RLLE. However, our RLTSA algorithm can achieve extraction robust local information and denoise in the robust PCA simultaneously, thus it can obtain better performance than RLLE on data set corrupted by a large amount of noise. This demonstrates the effectiveness of our RLTSA algorithm for noise manifold learning.

5.2. Rendered face data set

To illustrate the effectiveness of our RLTSA algorithm on high-dimensional real world data, we conduct experiments on rendered face data set which is another benchmark data set used by many manifold learning algorithms. This data set consists of 698 face images with 64×64 pixels collected under different poses and light conditions. In our experiments each image is represented as a 4096-dimensional vector. To generate noise images, we first randomly select 70 (≈10%) images and for each selected image we change the value of randomly selected 410 pixels (≈10%) by inverting each value (i.e., pixel value $v$ is replaced by $1-v$). Fig. 6 shows 10 original face images and their corresponding noise images. The 2-D embedding result of RLTSA is shown in Fig. 7. One can see from it that the poses and light of embedding images (including noise images) vary smoothly, this means that our RLTSA algorithm can recover the intrinsic manifold structure well in the presence of noise.

To further quantitatively evaluate the performance of low dimensional embedding obtained by LTSA, RLLE and RLTSA, we use Standard Diversity (SD) introduced in [22]. The standard diversity mainly focuses on the shape of a manifold and is invariant to translation, scale change and symmetrical deformation. Assume $Z = [z_1, z_2, \ldots, z_n]$ are the expected embedding, and $Y = [y_1, y_2, \ldots, y_n]$ are the embedding obtained by manifold learning algorithm where $z_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}^d$. Then the standard diversity of $Y$ is computed as

\[
SD(Y) = \frac{\min_{z_i, z_j} \|Z - (cz_i + L z_j)\|_F}{\|Z - (1/n)\text{vec}(Z)^T\|_F}
\]  

(21)

For the rendered face data set, the expected 3-D embedding is the pose and light parameter matrix $P$, which is also provided together with the face images. Thus we easily compute the standard diversities of the embeddings obtained by different algorithms. Fig. 8 plots the standard diversity of the 3-D coordinates computed by LTSA, MLTSA, RLLE and RLTSA under different neighborhood sizes, respectively. One can clearly see that our RLTSA algorithm leads to the smallest standard diversity, this means our RLTSA can recover the pose and light parameter with higher accuracy even in the presence of noise. However, the standard diversity of LTSA and MLTSA is close to 1, which means that in the presence of noise, MLTSA and LTSA cannot recover the intrinsic parameters of the data set.

5.3. Wood texture data set

Finally, we apply LTSA and RLTSA to wood texture images from the USC-SIPI image database. The images we used in our

---

**Fig. 4.** Results of four algorithms on 1500 data points sampled from helix curve where we select 150 data points at random and impose uniformly distributed noise on them such that the minimal distance between clean points and noise points is larger than 0.5. The $x$-axis in (b)–(e) indicates the 1-D embedding coordinates and the $y$-axis indicates the generating parameter of data set. (a) 1500 data points with 150 noise points, black points are noise points and colored points are clean points; (b) LTSA result; (c) MLTSA result; (d) RLLE result; (e) RLTSA result. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

---

3 http://isomap.stanford.edu

4 http://sipi.usc.edu/services/database
Fig. 5. Results of RLLE and RLTSA on Gaussian noise corrupted data sets sampled from Swiss Roll manifold. The data set contains 1500 points where we select different number of points at random and impose Gaussian noise with mean $m=0$ and standard deviation $\sigma=2$ on them. (a) RLLE result, 150 noise points; (b) RLTSA result, 150 noise points; (c) RLLE result, 300 noise points; (d) RLTSA result, 300 noise points; (e) RLTSA result, 600 noise points; (f) RLTSA result, 750 noise points.

Fig. 6. Ten face images and their corresponding noise images.

Fig. 7. 2-D embedding result of RLTSA algorithm. The “.” represents clean images, and red “*” represent noise images. Images correspond to the circled points linked by solid line. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Fig. 8. The standard diversity of four algorithms under different neighborhood size.
experiments are rotated wood texture images of three different rotation angles (30°, 90°, 120°). The original images are of size 512×512, we crop it into size of 300×300, then divide each of the images into 600 partially overlapping blocks of size 64×64. Thus we obtain 1800 wood texture images of 4096-dimensions. To generate noise images, we randomly choose 180 (10%) images (60 from each rotation angle), and for each selected image we change values of a randomly selected block of size 16×16 by replacing with the minimal value in that image.

Fig. 9 shows 15 noise images (5 images for each rotation angle) and their original images. The 2-D embedding results of LTSA and RLTSA are shown in Fig. 10. As can be seen, RLTSA can improve the performance of LTSA on noise data set. In the presence of noise, LTSA no longer preserves the separation between clusters, while our RLTSA can still preserve the separation between clusters well.

6. Conclusion

In this paper, a robust version of LTSA algorithm called RLTSA is proposed. RLTSA first performs local robust PCA instead of standard SVD to obtain local tangent coordinates; then using the result of robust PCA, RLTSA selects neighborhoods that are well approximated by the local coordinates, and distinctions are made on “clean” data points and “noise” points via giving them different alignment weights. Finally, RLTSA minimizes the local alignment error sum of selected neighborhoods to obtain embedding result. Extensive experiments on artificial data sets and real world data sets demonstrate the effectiveness of our RLTSA algorithm when dealing with noise corrupted data sets. Moreover, the mechanism dealing with different types of neighborhoods and different data points can be used in other manifold learning algorithms to make them robust.

![Fig. 9. Rotated wood texture images with different rotation angles and their noise corrupted images.](image-url)

![Fig. 10. 2-D embedding results of LTSA and RLTSA algorithms on rotated wood texture image data set. (a) Result of LTSA on clean image set; (b) result of RLTSA on noise data set; (c) result of LTSA on noise data set; (d) local amplified result of panel (c).](image-url)
References


Yubin Zhan received his B.Sci. and M.Eng. in School of Mathematics and Statistics from Wuhan University and Computer School from National University of Defense Technology in 2004 and 2006, respectively. And now he is a Ph.D. candidate in Computer School, National University of Defense Technology. His main research interests include machine learning, data mining and pattern recognition, especially application of manifold learning in image recognition.

Jianping Yin received his M.S. degree and Ph.D. degree in Computer Science from the National University of Defense Technology, China, in 1986 and 1990, respectively. He is a full professor of computer science in the National University of Defense Technology. His research interests involve artificial intelligence, pattern recognition, algorithm design, and information security.