J Intell Inf Syst (2009) 33:3–22
DOI 10.1007/s10844-008-0055-6

A new approach to discover interlacing data structures in high-dimensional space

Tao Ban · Changshui Zhang · Shigeo Abe

Published online: 26 May 2008

Abstract The discovery of structures hidden in high-dimensional data space is of great significance for understanding and further processing of the data. Real world datasets are often composed of multiple low dimensional patterns, the interlacement of which may impede our ability to understand the distribution rule of the data. Few of the existing methods focus on the detection and extraction of the manifolds representing distinct patterns. Inspired by the nonlinear dimensionality reduction method ISOmap, in this paper we present a novel approach called Multi-Manifold Partition to identify the interlacing low dimensional patterns. The algorithm has three steps: first a neighborhood graph is built to capture the intrinsic topological structure of the input data, then the dimensional uniformity of neighboring nodes is analyzed to discover the segments of patterns, finally the segments which are possibly from the same low-dimensional structure are combined to obtain a global representation of distribution rules. Experiments on synthetic data as well as real problems are reported. The results show that this new approach to exploratory data analysis is effective and may enhance our understanding of the data distribution.

Keywords Interlacing data structures · High-dimensional space · Real world dataset

T. Ban (✉)
Information Security Research Center, National Institute of Information and Communications Technology, Tokyo 184-8795, Japan
e-mail: bantao@nict.go.jp

C. Zhang
Department of Automation, Tsinghua University, Beijing 100-083, China
e-mail: zcs@mail.tsinghua.edu.cn

S. Abe
Graduate School of Science and Technology, Kobe University, Kobe 657-8501, Japan
e-mail: abe@eedept.kobe-u.ac.jp

1 Introduction

The goal of exploratory data analysis is to present a dataset in a form that is easily understandable, while preserving as much of the essential information in the data as possible (Jain and Dubes 1988). There are two well known approaches to simplify the problem and achieve the goal of illustrating hidden structures for a given dataset. One is to reduce the number of data items, the other is to reduce the dimensionality of the data.

Data clustering algorithms, i.e. partitional clustering and hierarchical clustering, take the first approach to yield a data description in terms of groups of data points that possess strong internal similarities (Duda et al. 2001). However, clustering methods often fail to detect delicate nonlinear structures in data due to two reasons: the neglect of intraclass topological structures and the overall usage of Euclidean distances.

With the second approach, linear dimensionality reduction methods such as Principal Component Analysis (PCA) (Jolliffe 1986) and Independent Component Analysis (Hyvarinen and Oja 2000), try to reduce statistical redundancy between the components of high dimensional data and enable a lower-dimensional representation without significant loss of information. However, when a nonlinear problem is involved, they always tend to overestimate the intrinsic dimensionality of the dataset. Unlike linear dimensionality reduction methods, their nonlinear alternatives manage to build low-dimensional representations while preserving the intrinsic geometry of the data. Thus nonlinear methods including five-layer autoassociators (Bourlard and Kamp 1988), ISOmap (TenenBaum et al. 2000), and Local Linear Embedding (LLE) (Roweis and Saul 2000) are expected to identify the underlying global structures in high dimensional input space.

The self-organizing maps (Kohonen 2001) are a special class of algorithms that can be used to reduce the amount of data by clustering while projecting the data onto a lower-dimensional display simultaneously. Typical applications are visualization of process states or financial results by representing the central dependencies within the data on the map.

Besides the nonlinearity, another cause of complexity in real world application data is the interlacement of patterns. Interlacement impedes the discovery of global structures and has brought a great challenge to data exploratory methods. Identification of these usually low dimensional patterns can help us obtain in-depth knowledge of the data, and more specifically, give guidance to subsequent data processing, e.g. dimensionality reduction and data visualization. It can also bring great convenience to successive analysis such as feature extraction, pattern matching, knowledge discovery, etc. Despite this significance, few of the existing methods focus on discovering these patterns from a mass of complicated input data.

In this paper we propose a new approach called Multi-Manifold Partition (MMP) to discover and separate manifolds which represent low dimensional patterns (or distribution rules) in high dimensional data space. We have borrowed the same basic ideas from ISOmap. In order to preserve the intrinsic geometry of the data, ISOmap first computes geodesic distances along a manifold as sequences of hops between neighboring points instead of Euclidean distances. It then applies classical Multidimensional Scaling (Cox and Cox 1994) to the matrix of geodesic distances to construct an embedding of the data in a low-dimensional Euclidean space. In

the algorithm presented here, a neighborhood graph is first built to capture the intrinsic topological structure of the input data. Then the intrinsic dimensionality of each point is estimated based on local PCA (Fukunaga and Olsen 1971). Finally, the neighboring nodes with uniform intrinsic dimensionality connected by the neighborhood graph are united into segments of manifolds which are named as primary structures here. (See Section 2 for a detailed definition.) Finally, combination of the primary structures possibly from the same manifold can lead to global low-dimensional structures (strokes), referred to as substructures afterwards.

The rest of the paper is organized as follows. In Section 2 we state the problem of MMP and give a brief review of recently proposed dimensionality estimation methods as well as some definitions and denotations. Section 3 specifies the proposed MMP algorithm which is implemented through two major subroutines: the primary structure searching algorithm and the primary structure joining algorithm. In Section 4 we show some experimental results on some simulated datasets as well as datasets derived from real world applications. Discussions are given in Section 5. Finally, Section 6 concludes the paper.

2 Foundations of the algorithms

For a dataset formed by the interlacement of a bunch of distinct patterns, it is interesting to discover the underlying structures which may represent easy to understand distribution rules in the data. Figure 1a gives an example of a mixed structure composed of two simple substructures, namely a plane and an S-like manifold. The plane can simply be regarded as 2D by a linear method, while a nonlinear dimension reduction method can find the intrinsic two-dimensionality of the manifold. In real world problems, such interlacing simple patterns can lead to troublesome high dimensionality as their number increases. Unfortunately, even nonlinear projection methods fail to discover the global structure, to say nothing of identifying the distinct low-dimensional patterns. Here we face the long time problem of intrinsic dimensionality estimation and address the novel problem of identification of interlacing patterns. In this section, we will give a brief review of intrinsic dimensionality estimation and the second problem will be detailed in the next section.

2.1 Intrinsic dimensionality estimation methods

The intrinsic or topological dimensionality (ID) of a dataset in d-dimensional space refers to the minimum number of "free" parameters needed to generate the dataset (Jain and Dubes 1988). Knowledge of the intrinsic dimensionality is important in order to determine the number of features necessary to represent the data. There are mainly two primary approaches for intrinsic dimensionality estimation. The first class of methods are global approaches which aim to analysis the global property of the dataset. The swarm of points are unfolded or flattened in the lower dimensional space. Methods like PCA and ISOmap belong to the class of global approaches. The second class of methods which are called local approaches includes the methods in Fukunaga and Olsen (1971) and Pettis et al. (1979) which try to estimate the intrinsic dimensionality directly from information in the neighborhood of data points without

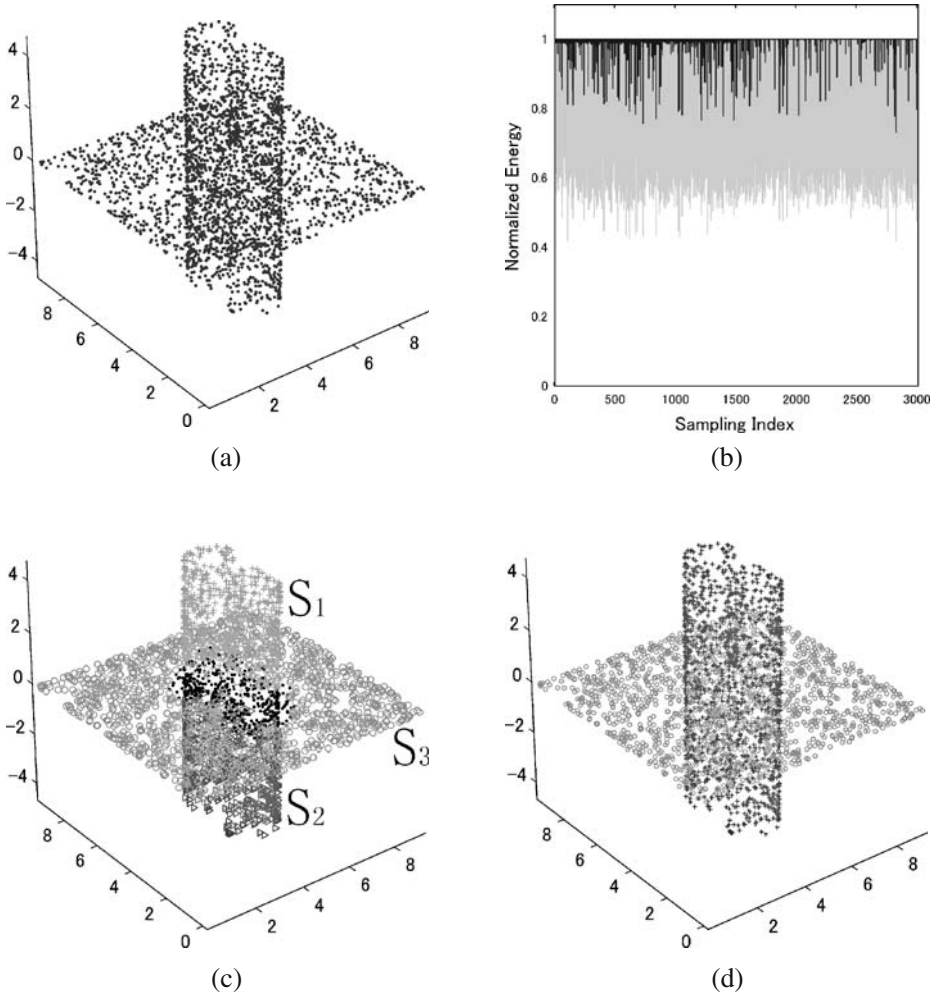


Fig. 1 Experiment I on 3D manifolds. **a** A plane and a manifold in 3D space. **b** Normalized energy distribution for local PCA. **c** Primary structure searching result. **d** Primary structure joining result

projecting the data points to a lower dimensional space. As we have mentioned, global approaches fail to estimate the intrinsic dimensionality of a dataset made up of mixed patterns because of the ununiformity of dimensionalities over different local regions. In this paper, we take a local approach, more specifically, the local PCA—introduced in (Fukunaga and Olsen 1971) and developed by Kambhatla and Olsen (Kambhatla and Leen 1997) and Verveer and Duin (1995), to determine the dimensionality of the local surface a data point resides in. Recent research on intrinsic dimensionality estimation from a local perspective can be found in (Camastra and Vinciarelli 2002; Costa and Hero 2004; Kégl 2003; Levina and Bickel 2005), where fractal numbers are yield by the algorithms to indicate the complexity of the data distribution.

The algorithm of Fukunaga and Olsen for estimating the intrinsic dimensionality is based on PCA. It is assumed that the intrinsic dimensionality of a dataset can be computed by dividing the dataset into small regions where the surfaces the data points locate are approximately linear. Obviously, the denser these points the more accurate the estimation provided by the PCA. The algorithm is divided into two steps:

(1) Partition the data space into distinct regions. This step is usually accomplished by vector quantization and so the method is called VQPCA;
(2) Estimate the intrinsic dimensionalities in separated local regions using PCA.

For a dataset with samples not in a uniform dimensional subspace, vector quantization may be misleading because data points at the edge of a Voronoi cell may have dimensionalities different from those of the points in the center. Unlike VQPCA, which attempts to find regions of low dimensionality and thus relies heavily on the partitioning algorithm, we use local PCA to determine the dimensionality of each data point. So in our approach, we circumvented the first step and simply build the K nearest neighbor regions to find the intrinsic dimensionality of the surface a data point resides in. The following section gives a more precise definition of K nearest neighbor regions.

2.2 Definitions and denotations

This section gives some definitions and denotations of the terms used in this paper. To make it more explicit, data points in a dataset are denoted as vectors in later discussion.

Definition 1 (K nearest neighbor region of vector v_i) Set $\omega_i = \{v_i, v_{i1}, .., v_{iK}\}$ is called the K nearest neighbor region of vector v_i, $(i = 1, 2, .., n)$, where n is the number of vectors in the dataset, and $v_{il}(l = 1, 2, .., K)$ denotes the lth nearest neighbor of v_i.

Definition 2 (f dimensionality) The f dimensionality of a dataset is the smallest dimensionality that can be used by method f to represent the dataset without too much loss of information. The **PCA dimensionality** of a vector v_i is defined as the PCA dimensionality of its K nearest neighbor region ω_i.

Generally speaking, data obtained from real world problems are always polluted by noises. So f dimensionality denotes the dimensionality by which method f can represent the data without significant loss of information, that is, within some reconstruction error criterion. The reconstruction error criterion ϵ is usually required to be smaller than some positive threshold near zero.

Given a dataset, different methods may find different dimensionalities. For example, for the manifold in Fig. 1a, PCA will find a dimensionality of three while ISOmap gives a dimensionality of two. For the K nearest neighbor region ω_i of a vector v_i, its PCA dimensionality equals its ISOmap dimensionality, because the geodesic distances between pairwise points equal the Euclidean distances (Cox and Cox 1994; TenenBaum et al. 2000). In the remainder of this paper, without special notification, the dimensionality of a local region indicates its PCA dimensionality.

Definition 3 (neighborhood graph) A neighborhood graph is a directed weighted graph $G = (V, E)$, where V is the set of vectors in the set and E is the set of edges from arbitrary vector v_i in V to the vectors in its K nearest neighbor region. The weight of an edge is the Euclidian distance between its starting vector and ending vector.

Two simple approaches to construct a neighborhood graph are to connect each vector to all vectors within some fixed radius, or to all of its K nearest neighbors. In our experiments, we take the second approach. More discussions on neighborhood graph construction is given in Section 5.

Definition 4 (primary structure) A dataset is defined as a primary structure when it satisfies all of the following conditions:

(1) Vectors in the set are directly connected to each other on the neighborhood graph.
(2) PCA dimensionalities of all vectors in the set are uniform.
(3) It is the parent set of all the sets that satisfy the first two conditions.

A **substructure** is composed of one or more equal dimensional primary structures. For example, the S-like manifold in Fig. 1c is divided into two segments, i.e. S_1 and S_2, by the overlapping region of the plane and the manifold. Each of them is called a primary structure, while the manifold itself is called a substructure. Evidently, S_1 and S_2 share some common characteristics. Firstly, each of them is a connected set in a topological sense. Secondly, the PCA dimensionalities of these two segments are no higher than that of their parent structure—the S-like manifold.

3 Algorithms

In this section, we specify the proposed MMP algorithm, which is further divided into two subroutines: the primary structure searching algorithm and the primary structure joining algorithm.

3.1 Algorithm 1: primary structure searching algorithm

The main idea of the primary structure searching algorithm is: First estimate the dimensionalities of all the vectors (i.e. the PCA dimensionalities of all K nearest neighbor regions). Then unite the vectors that not only have the same dimensionality but also are connected to one another on the neighborhood graph to form a primary structure. The complete algorithm as show in Table 1 has four steps. First, the neighborhood graph G is constructed for the input dataset $\mathcal{V}$. Then, the K-nearest neighbor regions $\Omega = \{\omega_1, \omega_2, .., \omega_n\}$ are formed according to the connectivity on the neighborhood graph. And then, the dimensionalities of regions ω_i, $(i = 1, 2, .., n)$ are estimated, here the local PCA based function is assumed. Finally, iterate the following joining process until no elements in Ω can be joined: for two arbitrary sets ω_i and ω_j $(i \neq j)$, if they have the same dimensionality and are directly connected

Table 1 Primary structure searching algorithm

Algorithm		
0	Inputs: $\mathcal{V}$, K;	
1	$G \leftarrow$BuildGraph($\mathcal{V}$, K);	// graph construction
2	$\Omega \leftarrow \{\omega_1, \omega_2, .., \omega_n\}$, $\omega_i \leftarrow K$Neighbor(G, v_i);	// get K-nearest neighbor sets
3	$d_i \leftarrow$ IntrinsicDim(ω_i), $i = 1, \cdots, n$;	// intrinsic dimension estimation
4	**do**	
5	**if** $d_i = d_j$ **and** Connected(G, v_i, v_j)	
6	$\omega_i \leftarrow \omega_i \cup \omega_j$;	// join two sets
7	$\Omega \leftarrow \Omega - \omega_j$;	
8	$d_i \leftarrow d_i$;	// preserve the dimensionality
9	**end if**	
10	**until** $\lvert\Omega\rvert \rightarrow C$	// iterate until converge
11	**return** Ω;	

in G, then let $\omega_i = \omega_i \cup \omega_j$ and delete ω_j from Ω. The dimensionality of ω_i is preserved.

With the primary structure searching algorithm, we can get the primary structures which represent the segments of distribution rules in the dataset. Here we note the set of primary structures as $\mathcal{S} = \{S_1, S_2, .., S_m\}$, where m is the number of primary structures in $\mathcal{S}$. Usually, the dimensionalities of the primary structures in $\mathcal{S}$ vary from one to another. There is some possibility that the estimated dimensionality of a primary structure is higher than its intrinsic dimensionality. This is potentially caused by two factors: first, the noises in the data are too loud to be ignored and leads to increased estimated dimensionality; second, a vector residing in the overlapping region of two or more lower dimensional manifolds will also yield a higher dimensionality.

3.2 Primary structures joining criteria

A real world dataset is often segmented into too many pieces by the primary structure searching algorithm. Two causes may lead us to such an embarrassment. On one hand, some higher dimensional primary structures are formed in overlapping regions of two or more lower dimensional manifolds. On the other hand, a single substructure representing a global topological structure may be separated into several parts by the overlapping regions. Here we give an algorithm to form substructures with global meaning. Two or more primary structures can be united into one if all of the following conditions are satisfied:

(Rule 1) They have identical dimensionality.
(Rule 2) They are geometrically close according to some proximity measure, as specified by the close neighbor criterion.
(Rule 3) They can be smoothly joined into one structure, as specified by the mutual representation criterion.

The first condition is apparent, hence we only discuss the other two conditions.

3.2.1 Close neighbor criterion

To illustrate the condition in Rule 2, we introduce the concept of geodesic distance (TenenBaum et al. 2000). The **geodesic distance** between two vectors v_i and v_j, noted by $g_d(i, j)$, is the length of the shortest path from v_i to v_j on the neighborhood graph G. The geometric distance between two primary structures S_p and S_q in $\mathcal{S}$, is define as

$$g_d(S_p, S_q) = \min_{v_i \in S_p, v_j \in S_q} g_d(v_i, v_j). \tag{1}$$

Primary structures S_p and S_q are **closely neighboring** when $g_d(S_p, S_q) < \delta$, where δ is a preset threshold parameter. Note that the appropriate δ value may vary greatly depending on the data distribution of different applications. However, the averaged edge length of the neighborhood graph G which represents the vicinity of nearest neighbors is a good reference to set the δ parameter.

3.2.2 Mutual representation criterion

Primary structures that satisfy the first two conditions may not be simply united. Take the dataset in Fig. 1c as an example. Although S_1 and S_3 satisfy both conditions of Rule 1 and Rule 2, a more reasonable combination is between S_1 and S_2. The reason is that we intuitively tend to unite two patterns with a continuous and smooth joint. So the third criterion is proposed to measure the smoothness of the joint.

Here, we generalize the concept of **closely opposed pairs** in Sklansky and Wassel (1981). For two primary structures S_p and S_q, a pair of vectors v_i, v_j, ($v_i \in S_p$, $v_j \in S_q$) are **closely opposed** if and only if the geodesic distance between v_i and v_j satisfies

$$d_g(v_i, v_j) = \min_{v_k \in S_q} d_g(v_i, v_k) = \min_{v_l \in S_p} d_g(v_l, v_j). \tag{2}$$

For two primary structures S_p and S_q, the set of closely opposed pairs, C, can be obtained by calculation of geodesic distances between all pairs of vectors.

Let Σ_i be the covariance matrix of ω_i, $U = \{u_1, u_2, .., u_d\}$ the associated eigenvectors of the d dominate eigenvalues of Σ_i, then the normalized reconstruction error e_h^i of a vector v_h can be computed as

$$e_h^i = \frac{\left|\left| v_h - \sum_{l=1}^{d} c_l \cdot u_l \right|\right|}{||v_h||}, \tag{3}$$

where $c_1, c_2, ..., c_d$ are the corresponding coefficients of v_h in the d-dimensional eigenspace. We say ω_j can be **linearly represented** by ω_i if the condition

$$\frac{\sum_{v_l \in \omega_j} e_l^i}{\sum_{v_h \in \omega_i} e_h^i} \leq \lambda, \tag{4}$$

where λ is a preset threshold parameter, is satisfied. Note that on the left hand side of the inequality, the numerator stands for the summed reconstruction error over set ω_j while the denominator stands for the summed reconstruction error over set

ω_i. When these two factors are comparative—measured by the λ parameter, the two sets probably reside in the same subspace. To be more clear, we use the averaged reconstruction error for two primary structures to evaluate the smoothness of the joint. Formally, the averaged reconstruction error to represent S_q by S_p is computed as

$$E_q^p = \frac{1}{|C|} \frac{\sum\limits_{\omega_j \subset C \cap S_q} \sum\limits_{v_l \in \omega_j} e_l^i}{\sum\limits_{\omega_i \subset C \cap S_p} \sum\limits_{v_h \in \omega_i} e_h^i}, \tag{5}$$

where $|C|$ is the cardinal number of the set of closely opposed pairs between primary structures S_p and S_q. For the closely opposed pairs in C, if the averaged reconstruction error is smaller than a preset λ value, we say S_p and S_q satisfy the **mutual representation criterion**.

3.3 Algorithm 2: primary structure joining algorithm

Before presenting the primary structure joining algorithm, we have to define the intersecting set between two sets. Let v_i and v_j be two arbitrary points in $\mathcal{V}$, G the neighborhood graph built from $\mathcal{V}$. Then the geodesic path between v_i and v_j, noted as $p_{i,j}$, is the set of points along the shortest path between v_i and v_j on G. The **intersecting set** between two sets $S_p, S_q \subset \mathcal{V}$ on a neighborhood graph G, noted as $S_{p,q}$, is the union of all the geodesic paths between all possible pairs of points which are drawn from S_p and S_q respectively. Formally,

$$S_{p,q} = \bigcup_{v_i \in S_p, v_j \in S_q} p_{i,j}. \tag{6}$$

Intuitively, $S_{p,q}$ are the set of points needed to connect two separated sets S_p and S_q into one, where the connectivity of all the points are defined with the neighborhood graph G. Note that $S_{p,q}$ not only comprises the connecting points but also includes the points in S_p and S_q as well. Consequently, an intersecting set may contain points which are already associated with some primary structures and a vector in the dataset may belong to multiple substructures returned by the primary structure joining algorithm.

Table 2 Primary structure joining algorithm

Algorithm		
0	Inputs: $\mathcal{S} = \{S_i\}, i = 1, \cdots, m$;	
1	**do**	
2	**if** Rule123(S_p,S_q) = *true*, $\forall S_p, S_q$	// qualified sets exist
3	$\mathcal{S} \leftarrow \mathcal{S} - S_q - S_p$;	
4	$\mathcal{S} \leftarrow \mathcal{S} \cup S_{p,q}$;	
5	$d_{p,q} \leftarrow d_p$;	// preserve the dimensionality
6	**end if**	
7	**until** ($\|\mathcal{S}\| \rightarrow C$)	// iterate until converge
8	**return** $\mathcal{S}$;	

With the definition of the joining conditions and the concept of intersecting set, we can present the following primary structure joining algorithm as listed Table 2. For the set of the primary structures $\mathcal{S}$, iterate the following joining process until no more sets can be joined: Let $S_{p,q}$ denote the intersecting set of primary structures S_p and S_q. For two sets S_p and S_q which satisfy Rule 1 through Rule 3, remove S_p and S_q from $\mathcal{S}$ and put $S_{p,q}$ into $\mathcal{S}$. The dimensionality of S_p is used for $S_{p,q}$.

4 Experiments

In this section, we test the proposed algorithm through a series of experiments. The application to both simulated datasets as well as datasets sampled from real world problems are demonstrated.

4.1 Computations

In order to use local PCA for dimensionality estimation we must eventually decide how many dominant eigenvalues exist in each local region, i.e. the threshold an eigenvalue obtained by each local PCA must exceed to indicate an associated intra-surface eigenvector. We adopted the Dα criterion from Fukunaga and Olsen (1971), which regards an eigenvalue μ_i as significant if

$$\frac{\mu_i}{\max_j(\mu_j)} > \alpha\%. \tag{7}$$

If no prior knowledge is available, different values of α have to be tested. Otherwise, knowledge of the largest noise component can be used to calculate α. In the experiments, we chose $\alpha_1 = \alpha_2 = \alpha_4 = 20, \alpha_3 = 10$ as the threshold to determine the dimensionalities of the local regions. And in all the three experiments, the threshold λ involved in primary structures joining algorithm is set to 2.

The number of nearest neighbors to generate the neighborhood graph and to form the vicinity set for local PCA based intrinsic dimensionality estimation, K, is set to $K_1 = K_2 = K_4 = 7$, and $K_3 = 11$ in the experiments. Some discussions on how to select this parameter is given in the discussion section. The δ parameter for the close neighbor criterion is set to 5 times the averaged edge length of G for all the evaluated datasets.

4.2 Experiment I

The dataset for experiment 1 is shown in Fig. 1a. The dataset contains two interlacing substructures in the 3D space: a 2D plane and an S-like manifold. The dataset is sampled from the manifolds with added Gaussian noise.

Figure 1b shows the normalized variance distributions in the directions of principal components of local regions. In the figure, the abscissa is the index of the vectors and the ordinate shows the normalized distribution of energy in local PCA eigenspaces. All the eigenvalues of each local region are normalized so that they sum up to 1. Distinct gray levels, from bottom to top, denote the relative energy contribution in the direction of the sorted dominant principal components for the

eigenspace. For almost all of the vectors, 1D projections can preserve only about 50% of their energy (white region in the figure); a majority of the vectors can be represented as 2D vectors by the first two eigenvectors with more than 95% of their energy preserved (white and gray regions in the figure); a few of the vectors have to be represented by 3D vectors to preserve a dominating part of their energy. According to the fact that most of the K nearest neighbor regions can be represented as 2D vectors by the local eigenvectors without much reconstruction error, local PCA returns the estimated dimensionality for these vectors as 2, and that of the rest vectors as 3.

Figure 1c shows the result of the primary structure searching algorithm by joining vectors with the same estimated dimensionality and are directly connected on the neighborhood graph G. Different primary structures extracted by the algorithm are shown as points with different markers and gray levels in the figure. Three 2D primary structures are labelled as S_1, S_2 and S_3, while the dimensionality of the black points is 3. It can be seen from the figure that, with the primary structure searching algorithm, the primary structures are correctly identified, though S_1 and S_2 are separated by the overlapping region of the two manifolds.

To apply the primary structure joining algorithm, the averaged reconstruction error E_j^i—reconstructing the closely neighboring K nearest neighbor regions in S_j by those in S_i—should be estimated. For the primary structures in experiment I, the averaged self reconstruction error $E_1^1 = 0.0355$, while the error using S_1 to reconstruct the remaining primary structures are $E_2^1 = 0.0417$ and $E_3^1 = 1.336$. Thus by setting the $\lambda = 2$, S_1 and S_2 are joined into one global substructure, i.e. the S-like manifold. Because the reconstruction errors $E_3^1 = 1.336$ and $E_3^2 = 1.511$ are far more larger than E_1^1, $S_{p,q}$ and S_3 can not be joined into one. The resultant substructures by joining the primary structures are shown in Fig. 1d. In the figure, S_1 and S_2 are joined to form the S-like manifold. As a post processing step, the vectors on the shortest paths connecting two arbitrary points in a substructure are assigned to the substructure.

4.3 Experiment II

In this experiment, we show the application of the proposed algorithm to analyze some 2D datasets. The datasets are sampled from interlacing curves in the 2D space. Points are represented as 2D vectors. At first, the primary structure searching algorithm is applied to identify the segments of curves. Then, the primary structure joining algorithm is used to discover the underlying global substructures.

The first dataset as shown in Fig. 2a-1 is drawn from interlacing straight lines. The data are subject to uniform distribution along the lines, with Gaussian noise added. Figure 2b-1 shows the normalized variance distributions in the directions of principal components of local regions. It can be found from the figure that for most of the vectors, the first eigenvalue for the local PCAs holds almost all the variance in the distribution. So that the estimated dimensionality for these vectors is 1. For the remaining vectors, the second eigenvalue is significant and the estimated dimensionality should be 2. In Fig. 2c-1, the 1D vectors are connected by the primary structure searching algorithm to form a number of primary structures which are presented by different markers. The 2D vectors with their nearest neighbors are shown as dark points in the figure. It is interesting to see that all these vectors fall

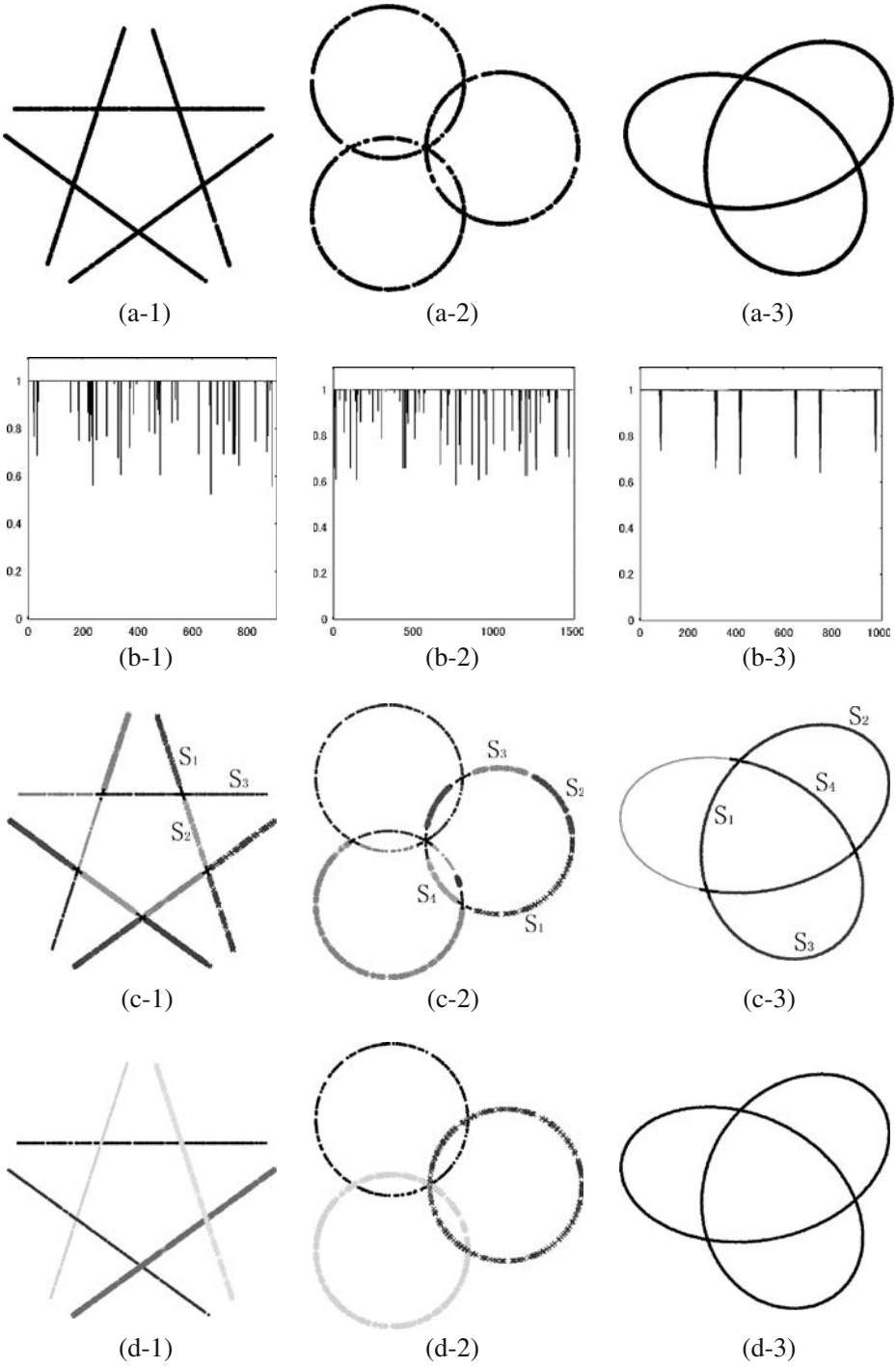


Fig. 2 Experiment II on 2D curves. **a** Input datasets. **b** Energy distributions of local PCAs. **c** Primary structure searching results. **d** Primary structure joining results

at the crossing point of the lines. Because vectors from multiple lines fall in their neighborhood, 2 dimensional subspaces are formed in the local regions. Then, the primary structure joining algorithm is applied to connect the primary structures from the same substructure. Apparently, in this case the underlining substructures are the line segments. To evaluate the mutual representation criterion, the reconstruction error between primary substructures are computed. For two primary structures from the same line segment, e.g. S_1 and S_2 in the figure, the relative reconstruction error is around 1 ($E_2^1 = 1.03E_1^1$). For two primary structures from different line segments, the error is much higher. For the case of S_1 and S_3, $E_3^1 = 35.7E_1^1$. In Fig. 2d-1, the joined substructures are shown. As expected, the substructures as represented by different markers are separated from each other. A following linear regression methods can be applied to further analyze the straight lines respectively.

The second dataset, shown in Fig. 2a-2, is drawn from multiple circles which are interlacing with each other in the 2D space. The data are distributed uniformly along the curves, with Gaussian noises added. Following the procedure for the first dataset, first see Fig. 2b-2 for the normalized variance distributions in the directions of principal components of local regions. Similar as the first dataset, for most of the vectors the estimated dimensionality is 1 and for the remaining vectors, the estimated dimensionality is 2. Then we can connect the directly connected 1D vectors on the neighborhood graph to form the primary structures. In Fig. 2c-2, the results of the primary structure searching algorithm is shown with different primary structures presented by different markers. Again, 2D vectors are formed at the crossing point of curves. This can prove the robustness of the local PCA method: by taking the local approach, the algorithm is adaptable to nonlinear problems which show linear property within local regions. Some interesting things different from the former dataset can be noticed in the figure. Although not separated by overlapping regions, the primary structures S_1, S_2, and S_3 which are sampled from the same circle are separated from each other. This is because that the three primary structures are not connected on the neighborhood graph G. There are two possible reasons: first, the K parameter to build G may be too small for the dataset; second, some clusters with comparatively small inter-cluster distances are formed. It can be found in the figure that the K parameter works well for the rest part of the data in this example, the problem is probably caused by isolated clusters. In Fig. 2d-2, the result of the primary structure joining algorithm is shown. Similar as the former example, primary structures which are from the same structures (the circles) are assigned correctly. Despite of the clusters, the algorithm successfully unites S_1, S_2, and S_3 into one substructure. In fact, the problem caused by the isolated clusters seems somewhat easier than that caused by the crossing points: the reconstruction error between S_1 and S_4 is $E_4^1 = 1.87E_1^1$, while the reconstruction error between S_1 and S_2 is $E_2^1 = 1.35E_1^1$.

While the first two datasets consist of multiple substructures, the third dataset is drawn from one smooth structure which is called a knot in the 3D space. Generally, a knot does not cross itself in the 3D space, so here we analyze the 2D projection of a knot structure as shown in Fig. 2a-3. In Fig. 2b-3, the normalized variance distributions in the local PCA regions are shown. From the variance distributions, we can judge that for most of the vectors the estimated dimensionality is 1 and for the remaining vectors it is 2. In Fig. 2c-3, the result of the primary structure searching algorithm is shown. Although the data are along the same curve, it is divided into

multiple primary structures by the crossing points. The estimated dimensionality of the crossing points (noted as dark points in the figure) is 2. Then the primary structure joining algorithm is applied to the primary structures to get the global substructure. The resultant substructure is shown in Fig. 4d-3. As expected, the whole knot curve is the only substructure returned as the result. Although primary structures S_1 and S_4 in Fig. 2c-3 are not directly connected because the reconstruction error $E_4^1 = 23.3E_1^1$ is much larger than the threshold of the mutual representation criterion, they are indirectly connected through S_3 to form a continuous substructure.

4.4 Experiment III

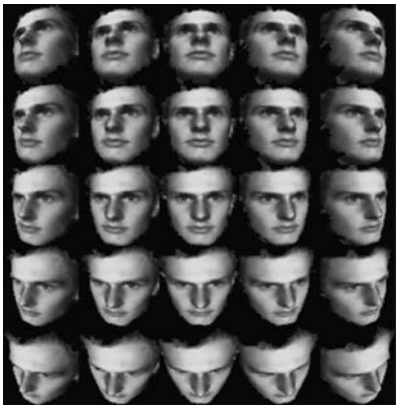
In Experiment I and II, we have demonstrated the application of the proposed algorithm to simulated datasets. The local PCA method gives a stable estimation of the dimensionality of the local neighborhood regions in spite of the added noises. The joining algorithm succeeded in combining primary structures which are drawn from the same underlining substructure. In the following, we test the algorithm on two datasets collected from real world applications.

The dataset in Experiment III consists of face images under different illumination and orientation conditions. Figure 3a shows some prototypes from the images group A, sampled from a rotating 3D face model with a fixed illumination environment and no occurrence of shadows. It includes 3721 images sampled uniformly over the feature space. Figure 2b shows some prototypes from image group B sampled from the same 3D face model. It contains a series of 181 images sampled during the procedure that a light source scans from left to right on the 3D model with a fixed posture. All of the images in these two groups are 64×64 in size and are converted to 4096D vectors as the system input.

Figure 3c shows the normalized variance distributions of local PCA regions. The arrow in the figure marks the divide of the two groups. It can be seen from the figure that, although the variance distribution is not as simple as the 2D cases, for the majority of the vectors in group A, the first two eigenvalues are much larger than the others that the PCA dimensionality of these vectors is 2. For the vectors in group A, the first eigenvalue is so significant that they can be treated as vectors of dimensionality 1. Other remaining vectors are assigned with a much higher dimensionality.

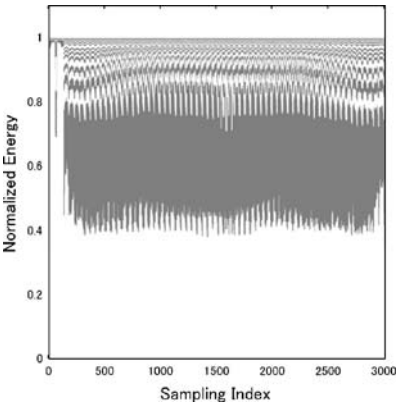
Figure 3d shows the result of the primary structure searching algorithm. For better visualization, we use ISOmap to project the dataset into a 2D embedding space. The substructure are shown as points of different markers. The prototypes shown in Fig. 3a and b are labelled as circles, with some of the corresponding images placed beside. In the figure, the dark gray points denote images belonging to two 1D primary structures. A light gray point denotes an image from a 2D primary structure. The remaining vectors with higher dimensionalities are marked as black points. From this example, it is easy to know that visualizing high dimensional datasets is not an easy task even with some advanced nonlinear manifold learning tools: Although these tools can help us to see the data in the 2D embedding space with a better understanding of the data distribution, the interlacing of the structures does impede further analysis of the data.

In Figs. 3e and f, the identified two substructures by the primary structure joining algorithm are shown respectively. The two substructures are projected into the 2D

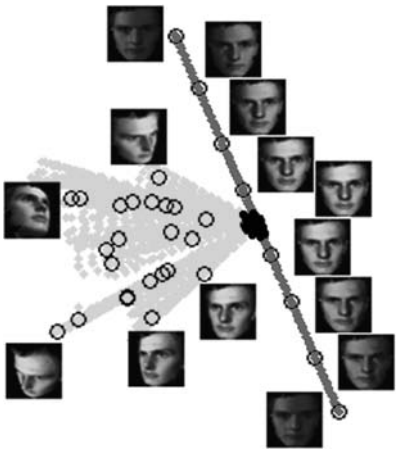


(a)

(b)



(c)

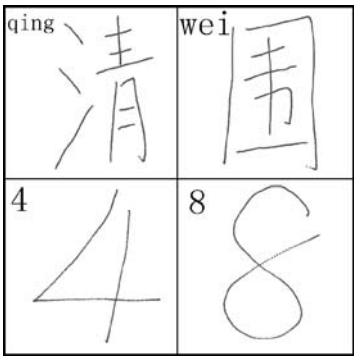


(d)

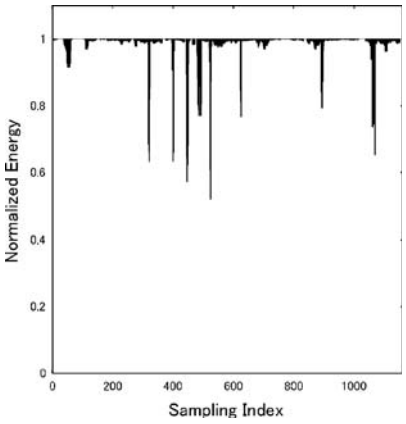
(e)

(f)

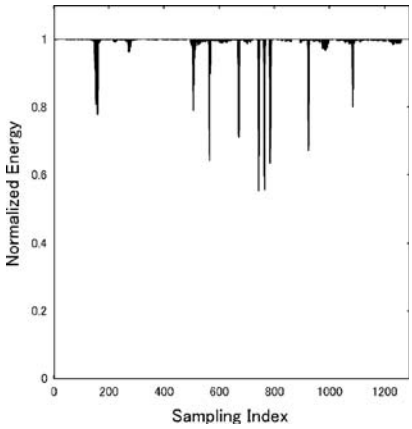
Fig. 3 Experiment III on 3D faces. **a** Prototypes from image group *A*. **b** Prototypes from group *B*. **c** Energy distributions of local PCAs. **d** Primary structure searching result. **e** The 2D substructure. **f** The 1D substructure



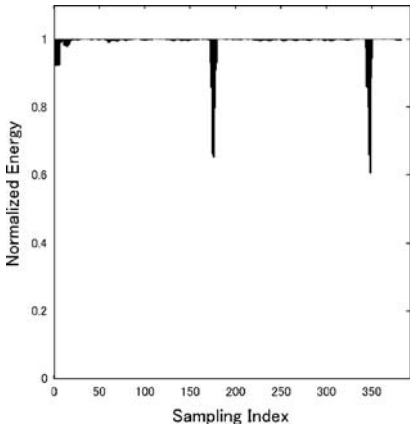
(a)



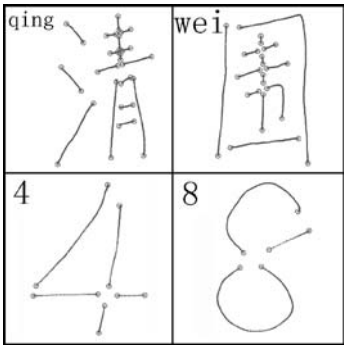
(b)



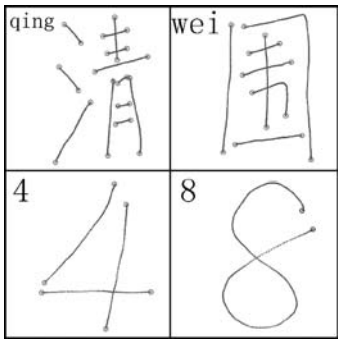
(c)



(d)



(e)



(f)

Fig. 4 Experiment IV on handwritten Chinese characters and digits. **a** Handwritten Chinese characters and digits. **b–d** Energy distributions of local PCAs for "qing", "wei", and "8". **e** Primary structure searching results. **f** Extracted strokes by primary structure joining algorithm. In **e** and **f**, the starting and ending points are emphasized by circles

space by ISOmap to ease the observation. In Fig. 3e, the substructure with a dimensionality of 2 is shown. This substructure mainly includes the images in group A, marked as gray points. Some of the images in group B marked as dark points are also included, because these points are along the shortest paths between points within the substructure. Figure 3f shows the 1D substructure in the 2D space. The substructure does not form a perfect line in the embedding space because of a small residual error. In addition to the gray points which are from image group B, some images from group A are also included in the substructure for they are on the shortest paths connecting the two separated primary structures. Comparing Fig. 3e and 3f with Fig. 3d, we can see that, by identifying the independent substructures in the dataset we can further enhance the understanding of the data distribution. Application of the manifold learning tools like LLE and ISOmap to the substructures will lead to simplified model of the data distribution. For this dataset, the 2D substructure can be modelled as a plane while the 1D structure can be analyzed using a curve fitting tool.

4.5 Experiment IV

In experiment IV, the proposed algorithm is evaluated by datasets with high level of noises, namely the handwritten digits and Chinese characters collected by a writing pad. Chinese characters are composed of elements called strokes. There are over 20 stroke patterns in frequently used characters. The shape and relative positioning of the strokes differentiate one character from others. Thus, strokes serve as the basic unit of Chinese character writing and stroke extraction is an essential preprocessing step for (optical) Chinese character recognition. However, for application where the time frame information is not available, interlacing of strokes always impedes our analysis (Chang and Wang 1994; Zeng and Liu 2006). In this experiment, we apply the proposed algorithm to solve stroke extraction problems.

The writing pad captures four features during the process a character is written on the pad: x and y coordinates of the pen, the time frame, and the exerted pressure. Here we only use the first two features as the input to the algorithm, thus the algorithm works in off-line mode with 2D input vectors. Some selected samples of handwritten Chinese characters and digits are shown in Fig. 4a. Each dataset consists of points collected during the writing of an individual character. It can be learned from the figure that the interlacing of strokes are very frequent for the Chinese characters. Note that loud noises are caused by manual input.

Figure 4b, c, and d give the variance distributions of local PCA analysis for “qing”, “wei” and “8” respectively. (The illustration for “4” is omitted for brevity.) From the energy distribution we can see that though loud noises are present, the local PCA can give a robust estimation of the dimensionality of local neighborhood regions. The dimensionality of a small part of the points is 2 while most of the points are with dimensionality of 1.

Figure 4e shows the extracted primary structures. For clarity, the 2D points are omitted from the figure. Comparing the characters in Fig. 4a we can see that, the points in the interlacing regions of strokes are identified as 2D points. Points at the smooth turnings, e.g. the stroke turning at the top right corner of “wei” which is smoothly transited, are identified as 1D points. Points at some sudden turnings, e.g. the ending hook of the referred stroke of “wei” and the bottom left corner of “4”, have a dimensionality of 2.

Then we can apply the primary structure joining algorithm to the primary structures denoting segments of strokes. Figure 4f presents the result of the algorithm. Stroke segments are integrated to strokes which constitute the characters. Except the ending points of some difficult strokes are dropped, the proposed MMP algorithm has given a quite good answer to the stroke extraction task. With an appropriate postprocessing algorithm to deal with the omitted ending points, MMP can be expected to serve as a preprocessing subsystem for a character recognition system.

5 Discussions

In this section we give some discussions on the implementation of the MMP algorithm.

f **dimensionality** Fukunaga and Olsen (1971) assumed that the intrinsic dimensionality of a dataset can be computed by dividing the set into small regions where the surfaces the vectors reside in are approximately linear. The intrinsic dimensionality is defined as the number of normalized eigenvalues that are larger than some threshold. Verveer and Duin have given a discussion in detail on this issue and proposed an improved method to estimate the dimensionality in Verveer and Duin (1995). In our implementation, we mainly follow the formulation in Fukunaga and Olsen (1971).

Construction of a neighborhood graph The choice of an appropriate number of nearest neighbors is critical to the algorithms we have presented and it is also an important problem in local PCA. With too few vectors in a local region, local PCA will not be a reasonable method to estimate the local dimensionality while too large a region size will lead to overestimation of the intrinsic dimensionality due to nonlinearity. While experiments show that for low dimensional manifolds, the number of neighboring vectors K is relatively small, theoretically little is known (Bruske and Sommer 1998). Our algorithm is base on the following assumptions:

(1) The number of nearest neighbors, K, is much greater than the intrinsic dimensionality of the local subspace that a vector resides in. This condition guarantees the effectiveness of local PCA.
(2) K is much smaller than the number of vectors in the dataset to make sure the local linearity condition is satisfied.

With these assumptions, we can get an efficient intrinsic dimensionality estimator which is a precondition for the primary structure searching and joining algorithms. Our experiments show that, although selection of appropriate K parameter is essential to the performance of MMP, a fairly wide value range with comparatively good results exists.

Noises Real data are always noisy and hence samples stemming from some low dimensional hypersurface will always contain noise orthogonal to the surface. Yet if the local region and the noises are small enough to support the linear assumption, local PCA will get the intrinsic dimensionality by identifying the dominating eigenvalues (Bruske and Sommer 1998). Some other works on intrinsic dimensionality are reported to be robust against noises (Kégl 2003; Raginsky and Lazebnik 2006). Incorporating these methods to the data analysis system will be explored in future work.

Computation costs The PCA of the $n \times n$ covariance matrix can be calculated in $O(n^3)$. The computation cost of local PCA of all the K nearest neighbor regions is $O(n \cdot K^3)$. When $K = n^{2/3}$, the computation costs are comparative, but generally, K is much smaller than n and our approach will outperform global PCA for dimensionality estimation of overall vectors.

The Dijkstra algorithm (Cormen et al. 1994) is used to compute the graph distance between pairwise vectors. The time complexity of this algorithm is $O(n^3)$. More efficient algorithms exploiting the sparse structure of the neighborhood graph can be found in Kumar et al. (1997).

From local scale to global scale Generally, the proposed primary structures joining algorithm can connect a number of primary structures with uniform dimensionality to a substructure of identical dimensionality which represents a underlying pattern. In some specific cases, connected segments may form end to end structures, e.g. curves can form circles and 2D manifolds may form sphere surfaces, and lead to increased complexity to the resultant substructure. However, such structures are still relatively simple and still meet our original intention to promote the data analysis.

6 Conclusions

In this paper, we have presented a novel approach named MMP to solve the problem of identifying topological structures for high dimensional data. The algorithm is implemented through the following steps. First, a neighborhood graph is built to capture the intrinsic topological structure of the input data. Second, the intrinsic dimensionality of a single point is estimated based on local Principle Component Analysis of its neighborhood regions and the neighboring nodes with uniform dimensionality are connected to form primary structures representing segments of distinct manifolds. Finally, combinations of the primary structures that are possibly from the same pattern leads to underlying global substructures which represents the distribution rules of the dataset.

Previous approaches to identify topological structures, say, clustering and the projection methods, can do well in cases where only a single global pattern is involved or delicate distribution rules are not considered respectively. When we need further knowledge about the structure of the data, our approach can be engaged to identify the underlying substructures which are ready for further analysis.

Anyway, data acquired from a real world problem may be so complex that all three approaches should all be involved. They may serve at different steps or at different scales to examine the data to gain in depth knowledge of the data distribution.

References

Bourlard, H., & Kamp, Y. (1988). Auto-association by multilayer perceptrons and singular value decomposition. *Biological Cybernetics, 59*, 291–294.

Bruske, J., & Sommer, G. (1998). An algorithm for intrinsic dimensionality estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 20*(5), 572–575.

Camastra, F., & Vinciarelli, A. (2002). Estimating the intrinsic dimension of data with a fractal-based method. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 24*(10), 1404–1407.

Chang, H. D., & Wang, J. F. (1994). A robust stroke extraction method for handwritten Chinese characters. *International Journal of Pattern Recognition and Artificial Intelligence, 8*(5), 1223–1239.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2001). *Introduction to algorithms* (2nd ed., pp. 595–601). Cambridge: MIT and McGraw-Hill.

Costa, J., & Hero, A. O. (2004). Geodesic entropic graphs for dimension and entropy estimation in manifold learning. *IEEE Transactions on Signal Processing, 52*(8), 2210–2221.

Cox, T., & Cox, M. (1994). *Multidimensional scaling.* London: Chapman & Hall.

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern classification* (2nd ed., pp. 517–556). New York: Wiley.

Fukunaga, K., & Olsen, D. R. (1971). An algorithm for finding intrinsic dimensionality of data. *IEEE Transactions on Computers, 20*, 176–183.

Hyvarinen, A., & Oja, E. (2000). Independent component analysis: Algorithms and applications. *Neural Networks, 13*, 411–430.

Jain, A. K., & Dubes, R. C. (1988). *Algorithms for clustering data.* Englewood Cliffs: Prentice Hall.

Jolliffe, I. T. (1986). *Principal component analysis.* New York: Springer-Verlag.

Kambhatla, N., & Leen, T. K. (1997). Dimension reduction by local principal component analysis. *Neural Computation, 9*(7), 1493–1516.

Kégl, B. (2003). Intrinsic dimension estimation using packing numbers. In T. G. Dietterich (Ed.), *Advances in neural information processing systems 14 (NIPS2002)*. Cambridge: MIT.

Kohonen, T. (2001). *Self-organizing maps, third extended edition, Springer series in information sciences* (Vol. 30). Berlin, Heidelberg, New York: Springer.

Kumar, V., Grama, A., Gupta, A., & Karypis, G. (1994). *Introduction to parallel computing: Design and analysis of algorithms* (pp. 257–297). Redwood City: Benjamin/Cummings.

Levina, E., & Bickel, P. J. (2005). Maximum likelihood estimation of intrinsic dimension. In L. K. Saul, Y. Weiss, & L. Bottou (Eds.), *Advances in neural information processing systems 17 (NIPS2004)*. Cambridge: MIT.

Pettis, K., Bailey, I., Jain, T., & Dubes, R. (1979). An intrinsic dimensionality estimator from near-neighbor information. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 1*, 25–37.

Raginsky, M., & Lazebnik, S. (2006). Estimation of intrinsic dimensionality using high-rate vector quantization. In Y. Weiss, B. Schölkopf, & J. Platt (Eds.), *Advances in neural information processing systems 18 (NIPS2005)*. Cambridge: MIT.

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

Sklansky, J., & Wassel, G. N. (1981). *Pattern classifiers and trainable machines* (pp. 112–113). New York: Springer-Verlag.

Tenenbaum, J. B., Silvam, V. de., & Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science, 290*, 2319–2323.

Verveer, P. J., & Duin, R. P. W. (1995). An evaluation of intrinsic dimensionality estimators. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 17*, 81–86.

Zeng, J., & Liu, Z. Q. (2006). Stroke segmentation of Chinese characters using Markov random fields. In *Proceedings of 18th international conference on pattern recognition (ICPR'06),* (pp. 868–871).