We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

A Fusion Scheme of Local Manifold Learning Methods

Xianglei Xing, Kejun Wang and Weixing Feng

Additional information is available at the end of the chapter

Abstract

Spectral analysis-based dimensionality reduction algorithms, especially the local manifold learning methods, have become popular recently because their optimizations do not involve local minima and scale well to large, high-dimensional data sets. Despite their attractive properties, these algorithms are developed based on different geometric intuitions, and only partial information from the true geometric structure of the underlying manifold is learned by each method. In order to discover the underlying manifold structure more faithfully, we introduce a novel method to fuse the geometric information learned from different local manifold learning algorithms in this chapter. First, we employ local tangent coordinates to compute the local objects from different local algorithms. Then, we utilize the truncation function from differential manifold to connect the local objects with a global functional and finally develop an alternating optimization-based algorithm to discover the low-dimensional embedding. Experiments on synthetic as well as real data sets demonstrate the effectiveness of our proposed method.

Keywords: dimensionality reduction, manifold learning

1. Introduction

Nonlinear dimensionality reduction (NLDR) plays an important role in the modern data analysis system, since many objects in our world can only be electronically represented with high-dimensional data such as images, videos, speech signals, and text documents. We usually need to analyze a large amount of data and process them, and however, it is very complicated or even infeasible to process these high-dimensional data directly, due to their high computational complexity on both time and space. Over the past decade, numerous manifold learning methods have been proposed for nonlinear dimensionality reduction. From methodology, these methods can be divided into two categories: global algorithms and local algorithms. Representative global algorithms contain isometric mapping [1], maximum variance unfolding

[2], and local coordinates alignment with global preservation [3]. Local methods mainly include Laplacian eigenmaps (LEM) [4], locally linear embedding (LLE) [5], Hessian eigenmaps (HLLE) [6], local tangent space alignment (LTSA) [7], local linear transformation embedding [8], stable local approaches [9], and maximal linear embedding [10].

Different local approaches try to learn different geometric information of the underlying manifold, since they are developed based on the knowledge and experience of experts for their own purposes [11]. Therefore, only partial information from the true underlying manifold is learned by each existing local manifold learning method. Thus, to better discover the underlying manifold structure, it is more informative and essential to provide a common framework for synthesizing the geometric information extracted from different local methods. In this chapter, we propose an interesting method to unify the local manifold learning algorithms (e. g., LEM, LLE, HLLE, and LTSA). Inspired by HLLE which employs local tangent coordinates to compute the local Hessian, we propose to utilize local tangent coordinates to estimate the local objects defined in different local methods. Then, we employ the truncation function from differential manifold to connect the local objects with a global functional. Finally, we develop an alternating optimization-based algorithm to discover the global coordinate system of lower dimensionality.

2. Local tangent coordinates system

A manifold is a topological space that locally resembles Euclidean space near every point. For example, around each point, there is a neighborhood that is topologically the same as the open unit ball in $\mathbb{R}^D$. The simplest manifold is a linear manifold, usually called a hyperplane. There exists a tangent space at each point of a nonlinear manifold. The tangent space is a linear manifold which locally approximates the manifold. Suppose there are N points $\{x_1,...,x_N\}$ in $\mathbb{R}^D$ residing on a smooth manifold $\mathcal{M} \subset \mathbb{R}^D$, which is the image of a coordinate space $\mathcal{Y} \subset \mathbb{R}^d$ under a smooth mapping $\psi : \mathcal{Y} \to \mathbb{R}^D$, where $d \ll D$. The mapping ψ is assumed as a locally isometric embedding. The aim of a NLDR algorithm is to acquire the corresponding low-dimensional representation $y_i \in \mathcal{Y}$ of each $x_i \in \mathcal{M}$ and preserve certain intrinsic structures of data at the same time. Suppose $\mathcal{M}$ is smooth such that the tangent space $T_x(\mathcal{M})$ is well defined at every point $x \in \mathcal{M}$. We can regard the local tangent space as a d-dimensional affine subspace of $\mathbb{R}^D$ which is tangent to $\mathcal{M}$ at x. Thus, the tangent space has the natural inner product induced by the embedding $\mathcal{M} \subset \mathbb{R}^D$. Within some neighborhood of x, each point $x \in \mathcal{M}$ has a sole closest point in $T_x(\mathcal{M})$, and therefore, an orthonormal coordinate system from the corresponding local coordinates on $\mathcal{M}$ can be associated with the tangent space.

A manifold can be represented by its coordinates. While the current research of differential geometry focuses on the characterization of the global properties of manifolds, NLDR algorithms, which try to find the coordinate representations of data, only need the local properties of manifolds. In this chapter, we use local coordinates associated with the tangent space to estimate the local objects over the manifold. To acquire the local tangent coordinates, we first perform Principal Component Analysis (PCA) [12] on the points in $\mathcal{N}(x_i) = \{x_i, x_{i_1},..., x_{i_k}\}$ that

is the local patch built by the point x_i and its k nearest neighborhoods, and get d leading PCA eigenvectors $V^i = \{v^i_1, v^i_2, \ldots, v^i_d\}$ which correspond to an orthogonal basis of $T_{x_i}(\mathcal{M})$ (the orthogonal basis can be seen as a d-dimensional affine subspace of $\mathbb{R}^D$ which is tangent to $\mathcal{M}$ at x_i). For high-dimensional data, we employ the trick presented by Turk and Pentland for EigenFaces [13]. Then, we obtain the local tangent coordinates $\mathcal{U}^i = \{0, u^i_1, \ldots, u^i_k\}$ of the neighborhood $\mathcal{N}(x_i)$ by projecting the local neighborhoods to this tangent subspace:

$$u^i_j = (V^i)^T (x_{i_j} - x_i) \tag{1}$$

An illustration of the local tangent space at x_i and the corresponding tangent coordinates system (i.e., the point x_{i_j}'s local tangent coordinate is u^i_j) is shown in **Figure 1**.
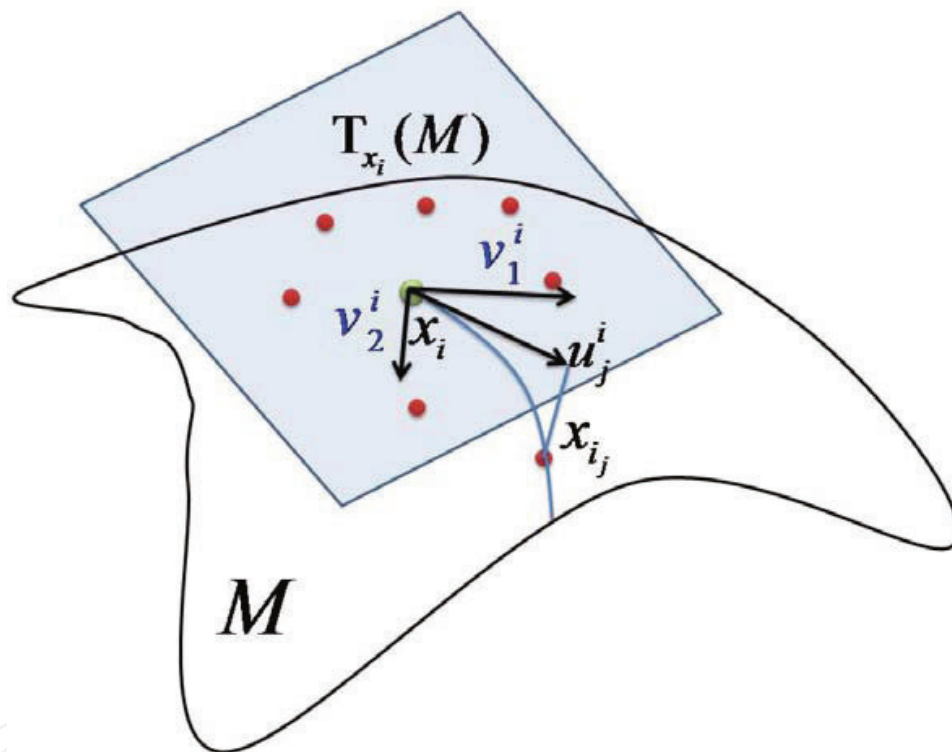

Figure 1. Local tangent space and tangent coordinates system.

3. Reformulations of LEM, LLE, HLLE and LTSA using local tangent coordinates

3.1. Reformulation of Laplacian eigenmaps

The method LEM was introduced by Belkin and Niyogi [4]. We can summarize the geometrical motivation of LEM as follows. Assume that we are searching for a smooth one-dimensional embedding $f : \mathcal{M} \rightarrow \mathbb{R}$ from the manifold to the real line so that data points near each together

on the manifold are also mapped close together on the line. Think about two adjacent points, $x,z \in \mathcal{M}$, which are mapped to $f(x)$ and $f(z)$, respectively, we can obtain that

$$|f(z)-f(x)| \leq \|\nabla_\mathcal{M} f(x)\| \|z-x\| + O(\|z-x\|^2) \tag{2}$$

where $\nabla_\mathcal{M} f$ is the gradient vector field along the manifold. Thus, to the first order, $\|\nabla_\mathcal{M} f\|$ provides us with an estimate of how far apart f maps nearby points. When we look for a map that best preserves locality on average, a natural choice to find f is to minimize [4]:

$$\Phi_{lap}(f) = \int_\mathcal{M} \|\nabla_\mathcal{M} f\|^2 = \int_\mathcal{M} \Delta_\mathcal{M}(f)f \tag{3}$$

where the integral is taken with respect to the standard measure over the manifold. Thus, the function f that minimizes $\Phi_{lap}(f)$ has to be an eigenfunction of the Laplace-Beltrami operator $\Delta_\mathcal{M}$, which is a key geometric object associated with a Riemannian manifold [14].

Suppose that the tangent coordinate of $x \in \mathcal{N}(x)$ is given by u. Then, the rule $g(u) = f(x) = f \circ \psi(u)$ defines a function $g : U \to \mathbb{R}$, where U is the neighborhood of $u \in \mathbb{R}^d$. With the help of local tangent coordinates, we can reduce the computation of the gradient vector $\nabla_\mathcal{M} f(x)$ on the manifold to the computation of the ordinary gradient vector on the Euclidean space:

$$\nabla_{tan} f(x) = \nabla g(u) = \left(\frac{\partial g(u)}{\partial u^1}, \cdots, \frac{\partial g(u)}{\partial u^d} \right)^T \tag{4}$$

where $u = (u^1, \ldots, u^d) \in \mathbb{R}^d$, and we keep up tan in the notation to make clear that it counts on the coordinate system in $T_x(\mathcal{M})$. For different local coordinate systems, although the tangent gradient vector will be different, the norm $\|\nabla_{tan} f(x)\|$ is inimitably defined such that equation (3) can be approximated by estimating the following functional:

$$\tilde{\Phi}_{lap}(f) = \int_\mathcal{M} \|\nabla_{tan} f(x)\|^2 dx \tag{5}$$

where dx stands for the probability measure on $\mathcal{M}$.

In order to compute the local object $\|\nabla_{tan} f(x)\|^2$, we first use the first-order Taylor series expansion to approximate the smooth functions $\{f(x_{i_j})\}_{j=1}^k, f : \mathcal{M} \to \mathbb{R}$, and together with Eq. (4), we have:

$$\begin{aligned} f(x_{i_j}) &= f(x_i) + (\nabla_{tan} f(x_i))^T (x_{i_j}-x_i) + O(\|x_{i_j}-x_i\|^2) \\ &= g(u_j^i) = g(0) + (\nabla_{tan} f(x_i))^T u_j^i + O(\|u_j^i\|^2) \end{aligned} \tag{6}$$

Over $\mathcal{U}^i$, we develop the operator $\alpha^i = [g(0), \nabla g(0)] = [g(0), \nabla_{tan} f(x_i)]$ that approximates the function $g(u_j^i)$ by its projection on the basis $U_j^i = \{1, u_{j_1}^i, \ldots, u_{j_d}^i\}$:

$$f(x_{i_j}) = g(u_j^i) = (\alpha^i)^T U_j^i \tag{7}$$

The least-squares estimation of the operator α^i can be computed by:

$$\underset{\alpha^i}{\operatorname{argmin}} \sum_{j=1}^{k} (f(x_{i_j}) - (\alpha^i)^T U_j^i)^2 \tag{8}$$

It is easy to show that the least-squares solution of the above object function is $\alpha^i = (U^i)^\dagger f^i$, where $f^i = [f(x_{i_1}), \ldots, f(x_{i_k})] \in \mathbb{R}^k$, $U^i = [U_1^i; U_2^i; \ldots; U_k^i] \in \mathbb{R}^{k \times (1+d)}$, and $(U^i)^\dagger$ denotes the pseudo-inverse of U^i. If we define a local gradient operator $G^i \in \mathbb{R}^{d \times k}$ which is constructed by the last d rows of $(U^i)^\dagger$, we have $\nabla_{tan} f(x_i) = G^i f^i$. Furthermore, the local object $\|\nabla_{tan} f(x_i)\|^2$ can be computed as:

$$\|\nabla_{tan} f(x_i)\|^2 = \nabla_{tan} f(x_i)^T \nabla_{tan} f(x_i) = (f^i)^T (G^i)^T G^i f^i \tag{9}$$

An unresolved problem in our reformulation is how to connect the local object $\|\nabla_{tan} f(x)\|^2$ with the global functional $\tilde{\Phi}_{lap}(f)$ in (5) and its discrete approximation. In Section 4, we will discuss this issue in detail.

3.2. Reformulation of locally linear embedding

The LLE method was introduced by Roweis and Saul [5]. It is based on simple geometric intuitions, which can be depicted as follows. Globally, the data points are sampled from a nonlinear manifold, while each data point and its neighbors are residing on or close to a linear patch of the manifold locally. Thus, it is possible to describe the local geometric properties of the neighborhood of each data point in the high-dimensional space by linear coefficients which reconstruct the data point from its neighbors under suitable conditions. The method of LLE computes the low-dimensional embedding which is optimized to preserve the local configurations of the data. In each locally linear patch, the reconstruction error in the original LLE can be written as:

$$\hat{\varepsilon}^i = \|x_i - \sum_{j=1}^{k} w_{i_j} x_{i_j}\|^2 \tag{10}$$

where $\{w_{i_j}\}_{j=1}^{k}$ are the reconstruction weights which encode the geometric information of the high-dimensional inputs and are constrained to satisfy $\sum_j w_{i_j} = 1$.

Since the geometric structure of the local patch can be approximated by its projection on the tangent space $T_{x_i}(\mathcal{M})$, we utilize the local tangent coordinates to estimate the local objects over the manifold in our reformulation framework. We can write the reconstruction error of each local tangent coordinate as:

$$\varepsilon^i = \|u_i - \sum_{j=1}^{k} w_{i_j} u_j^i\|^2 = \|\sum_j w_{i_j}(u_i - u_j^i)\|^2 = \sum_{jk} w_{i_j} w_{i_k} G_{jk}^i \tag{11}$$

where we have employed the fact that the weights sum to one, and G^i is the local Gram matrix,

$$G_{jk}^i = \langle(u_i - u_j^i),(u_i - u_k^i)\rangle \tag{12}$$

The optimal weights can be obtained analytically by minimizing the above reconstruction error. We solve the linear system of equations

$$\sum_k G_{jk}^i w_{i_k} = 1 \tag{13}$$

and then normalize the solution by $\sum_k w_{i_k} = 1$. Consider the problem of mapping the data points from the manifold to a line such that each data point on the line can be represented as a linear combination of its neighbors. Let $f(x_{i_1}),\dots,f(x_{i_k})$ denote the mappings of $u_1^i,\dots,u_k^i$, respectively. Motivated by the spirit of LLE, the neighborhood of $f(x_i)$ should share the same geometric information as the neighborhood of u_i, so we can define the following local object:

$$|\sigma_f(x_i)|^2 = |f(x_i) - \sum_{j=1}^{k} w_{i_j} f(x_{i_j})|^2 = (f^i)^T (W^i)^T W^i f^i \tag{14}$$

where $W^i = [1, -w_i] \in \mathbb{R}^{1\times(k+1)}$, $f^i = [f(x_i), f(x_{i_1}), \dots, f(x_{i_k})]$. The optimal mapping f can be obtained by minimizing the following global functional:

$$\mathcal{E}(f) = \int_{\mathcal{M}} |\sigma_f(x)|^2 dx \tag{15}$$

where dx stands for the probability measure on the manifold.

3.3. Reformulation of Hessian eigenmaps

The HLLE method was introduced by Donoho and Grimes [6]. In contrast to LLE that obtains linear embedding by minimizing the l_2 error in Eq. (10), the HLLE achieves linear embedding by minimizing the Hessian functional on the manifold where the data points reside. HLLE supposes that we can obtain the low-dimensional coordinates from the $(d+1)$-dimensional null-space of the functional $\mathcal{H}(f)$ which presents the average curviness of f upon the manifold, if the manifold is locally isometric to an open connected subset of $\mathbb{R}^d$. We can measure the functional $\mathcal{H}(f)$ by averaging the Frobenius-norm of the Hessians on the manifold $\mathcal{M}$ as [6]:

$$\mathcal{H}(f) = \int_{\mathcal{M}} \|H_f^{tan}(x)\|_F^2 dx \tag{16}$$

where H_f^{tan} stands for the Hessian of f in tangent coordinates. In order to estimate the local Hessian matrix, we first perform a second-order Taylor expansion at a fixed x_i on the smooth functions: $\{f(x_{i_j})\}_{j=1}^{k}, f : \mathcal{M} \to \mathbb{R}$ that is C^2 near x_i:

$$f(x_{i_j}) \approx f(x_i) + (\nabla f)^T (x_{i_j} - x_i) + \frac{1}{2}(x_{i_j} - x_i)^T H_f^i (x_{i_j} - x_i)$$
$$= g(u_j^i) = g(0) + (\nabla g)^T u_j^i + \frac{1}{2} u_j^{iT} H_f^i u_j^i + O(\|u_j^i\|^3)$$
(17)

Here, $\nabla f = \nabla g$ is the gradient defined in (4), and H_f^i is the local Hessian matrix defined as:

$$(H_f^i)_{p,q}(x) = \frac{\partial}{\partial u_p} \frac{\partial}{\partial u_q} g(u)$$
(18)

where $g : U \to \mathbb{R}$ uses the local tangent coordinates and satisfies the rule $g(u) = f(x) = f \circ \psi(u)$. In the second identity of Eq. (17), we have exploited the fact that $u_i^i = \langle V^i, x_i - x_i \rangle = 0$ [recall the computation of local tangent coordinates in Eq. (1)].

Over $\mathcal{U}^i$, we develop the operator β^i that approximates the function $g(u_j^i)$ by its projection on the basis $U_j^i = \{1, u_{j_1}^i, \ldots, u_{j_d}^i, (u_{j_1}^i)^2, \ldots, (u_{j_d}^i)^2, \ldots, u_{j_1}^i \times u_{j_2}^i, \ldots, u_{j_{d-1}}^i \times u_{j_d}^i\}$, and we have:

$$f(x_{i_j}) = g(u_j^i) = (\beta^i)^T U_j^i$$
(19)

Let $\beta^i = [g(0), \nabla g, h^i] \in \mathbb{R}^{1+d+d(d+1)/2}$, then $h^i \in \mathbb{R}^{d(d+1)/2}$ is the vector form of local Hessian matrix H_f^i over neighborhood $N(x_i)$. The least-squares estimation of the operator β^i can be obtained by:

$$\underset{\beta^i}{\operatorname{argmin}} \sum_{j=1}^{k} (f(x_{i_j}) - (\beta^i)^T U_j^i)^2$$
(20)

The least-squares solution is $\beta^i = (U^i)^\dagger f^i$, where $f^i = [f(x_1), \ldots, f(x_k)] \in \mathbb{R}^k$, $U^i = [U_1^i; U_2^i; \ldots; U_k^i] \in \mathbb{R}^{k \times (1+d+d(d+1)/2)}$, and $(U^i)^\dagger$ signifies the pseudo-inverse of U^i. Notice that h^i is the vector form of local Hessian matrix H_f^i, while the last $d(d+1)/2$ components of β^i correspond to h^i. Meanwhile, we can construct the local Hessian operator $H^i \in \mathbb{R}^{(d(d+1)/2) \times k}$ by the last $d(d+1)/2$ rows of $(U^i)^\dagger$, and therefore, we can obtain $h^i = H^i f^i$. Thus, the local object $\|H_f^{tan}(x_i)\|_F^2$ can be estimated with:

$$\|H_f^{tan}(x_i)\|_F^2 = (h^i)^T (h^i) = (f^i)^T (H^i)^T (H^i)(f^i)$$
(21)

3.4. Reformulation of local tangent space alignment

The method LTSA was introduced by Zhang and Zha [7]. LTSA is based on similar geometric intuitions as LLE. The neighborhoods of each data point remain nearby and similarly colocated in the low-dimensional space, if the data set is sampled from a smooth manifold. LLE constructs low-dimensional data so that the local linear relations of the original data are preserved, while LTSA constructs a locally linear patch to approximate the tangent space at the point. The coordinates provided by the tangent space give a low-dimensional representation of the patch. From Eq. (6), we can obtain:

$$f(x_{i_j}) = f(x_i) + (\nabla_{tan}f(x_i))^T u_j^i + O(\|u_j^i\|^2) \tag{22}$$

From the above equation, we can discover that there are some relations between the global coordinate $f(x_{i_j})$ in the low-dimensional feature space and the local coordinate u_j^i which represents the local geometry. The LTSA algorithm requires the global coordinates $f(x_{i_j})$ that should respect the local geometry determined by the u_j^i:

$$f(x_{i_j}) \approx \bar{f}(x_i) + L_i u_j^i, \tag{23}$$

where $\bar{f}(x_i)$ is the mean of $f(x_{i_j})$, $j = 1,\dots,k$. Inspired by LTSA, the affine transformation L_i should align the local coordinate with the global coordinate, and we can define the following local object:

$$|\kappa_f(x_i)|^2 = |(f^i)^T - \frac{1}{k}(f^i)^T ee^T - L_i U^i|^2, \tag{24}$$

where $f^i = [f(x_{i_1}),\dots,f(x_{i_k})]^T$, $U^i = [u_1^i; u_2^i; \dots; u_k^i]$, and e is a k-dimensional column vector of all ones. Naturally, we should seek to find the optimal mapping f and a local affine transformation L_i to minimize the following global functional:

$$\mathcal{K}(f) = \int_{\mathcal{M}} |\kappa_f(x)|^2 dx \tag{25}$$

Obviously, the optimal affine transformation L_i that minimizes the local reconstruction error for a fixed f^i is given by:

$$L_i = (f^i)^T \left(I - \frac{1}{k}ee^T\right)(U^i)^\dagger \tag{26}$$

and therefore,

$$|\kappa_f(x_i)|^2 = |(f^i)^T \left(I - \frac{1}{k}ee^T\right)(I - (U^i)^\dagger U^i)|^2, \tag{27}$$

Let $W^i = (I - (U^i)^\dagger U^i)^T (I - \frac{1}{k}ee^T)^T$, the local object $\kappa_f(x_i)$ can be estimated as:

$$|\kappa_f(x_i)|^2 = |(f^i)^T \left(I - \frac{1}{k}ee^T\right)(I - (U^i)^\dagger U^i)|^2 = (f^i)^T (W^i)^T (W^i)(f^i) \tag{28}$$

4. Fusion of local manifold learning methods

So far we have discussed four basic local objects: $\|\nabla_{tan}f(x)\|^2$, $|\sigma_f(x)|^2$, $\|H_f^{tan}(x_i)\|_F^2$, and $|\kappa_f(x_i)|^2$. From different perspectives, they depict the geometric information of the manifold. We look forward to collect these geometric information together to better reflect the geometric structure

of the underlying manifold. Notice that we can estimate these local objects under the local tangent coordinate system according to Eqs. (9), (14), (21), and (28), respectively. Taking stock of the structure of these equations, it is not hard to discover that we can fuse these local objects together under our proposed framework. Assume that there are M different local manifold learning algorithms, we can define the fused local object as follows:

$$LO_f(x) = \sum_{j=1}^{M} c_j LO_j(x) \tag{29}$$

where $\{c_j\}_{j=1}^{M}$ are the nonnegative balance parameters, $\{LO_j(x)\}_{j=1}^{M}$ are the local objects, such as $\|\nabla_{tan}f(x)\|^2$, $|\sigma_f(x)|^2$, $\|H_f^{tan}(x_i)\|_F^2$, and $|\kappa_f(x_i)|^2$, from different algorithms. It is worth to note that the other local manifold learning algorithms can also be reformulated to incorporate into our unified framework.

We employ the truncation function from differential manifold to connect the local objects with their corresponding global functional such that we can obtain a consistent alignment of the local objects to discover a single global coordinate system of lower dimensionality. The truncation function is a crucial tool in differential geometry to build relationships between global and local properties of the manifold. Assume that U and V are two nonempty subsets of a smooth manifold $\mathcal{M}$, where $\overline{V}$ is compact and $\overline{V} \in U$ ($\overline{V}$ is the closure of V). Accordingly, the truncation function [15] can be defined as a smooth function $s : \mathcal{M} \to \mathbb{R}$ such that:

$$s(p) = \begin{cases} 1, & p \in V \\ 0, & p \notin U. \end{cases} \tag{30}$$

The truncation function s can be discretely approximated by the 0-1 selection matrix $S^i \in \mathbb{R}^{N \times k}$. An entry of S^i is defined as:

$$(S^i)_{pq} = \begin{cases} 1, & p = N_i\{q\} \\ 0, & p \neq N_i\{q\}. \end{cases} \tag{31}$$

where $N_i = \{i_1, \ldots, i_k\}$ denotes the set of indices for the k-nearest neighborhoods of data point x_i. Let $f = [f(x_1), \ldots, f(x_N)] \in \mathbb{R}^N$ be a function defined on the whole data set sampled from the global manifold. Thus, the local mapping $f^i = [f(x_1^i), \ldots, f(x_k^i)] \in \mathbb{R}^k$ can be expressible by $f^i = (S^i)^T f$. With the help of the selection matrix, we can discretely approximate the global functional $\mathcal{G}(f)$ as follows:

$$\begin{aligned} \mathcal{G}(f) &= \int_{\mathcal{M}} LO_f(x)\, dx = \frac{1}{N} \sum_{i=1}^{N} LO_f(x_i) \\ &= \frac{1}{N} \sum_{i=1}^{N} (f^i)^T \left(\sum_{j=1}^{M} c_j L_j^i \right) f^i = f^T \left(\sum_{j=1}^{M} c_j P^j \right) f \end{aligned} \tag{32}$$

where $\{L_j^i\}_{j=1}^{M}$ are the local matrices such as $(G^i)^T G^i$, $(W^i)^T W^i$, $(H^i)^T H^i$, and $(W^i)^T W^i$ which are defined in Eqs. (9), (14), (21), and (28). $P^j = \frac{1}{N} \sum_{i=1}^{N} S^i L_j^i (S^i)^T$ is the alignment matrix of the j-th

local manifold learning method. The global embedding coordinates $Y = [y_1, y_2, \ldots, y_N] \in \mathbb{R}^{d \times N}$ can be obtained by minimizing the functional $\mathcal{G}(f)$. Let $y = f = [f(x_1), \ldots, f(x_N)]$ be a row vector of Y. It is not hard to show that the global embedding coordinates and the nonnegative weights $c = [c_1, \ldots, c_M]$ can be obtained by minimizing the following objective function:

$$\underset{Y,c}{\text{argmin}} \sum_{j=1}^{M} c_j^r Tr(YP^jY^T) \, s.t. \, YY^T = I; \, \sum_{j=1}^{M} c_j = 1, \, c_j \geq 0. \tag{33}$$

where the power parameter $r > 1$ is set to avoid the phenomenon that the solution to c is $c_j = 1$ corresponding to the minimum $Tr(YP^jY^T)$ over different local methods and $c_k = 0 (k \neq j)$ otherwise, since our aim is to utilize the complementary geometric information from different manifold learning methods.

We propose to solve the objective function [Eq. (33)] by employing the alternating optimization [16] method, which iteratively updates Y and c in an alternating fashion. First, we fix c to update Y. The optimization problem in Eq. (33) is equivalent to:

$$\underset{Y}{\text{argmin}} \, Tr(YPY^T) \, s.t. \, YY^T = I \tag{34}$$

where $P = \sum_{j=1}^{M} c_j^r P^j$. When c is fixed, we can solve the optimization problem [Eq. (34)] and obtain the global optimal solution Y as the second to $(d+1)$ st smallest eigenvectors of the matrix P. Second, we fix Y to update c. While Y is fixed, we can minimize the objective function [Eq. (33)] analytically through utilizing a Lagrange multiplier to enforce the constraint that $\sum_{j=1}^{M} c_j = 1$. And the global optimal c can be obtained as:

$$c_j = \frac{(1/Tr(YP^jY^T))^{1/(r-1)}}{\sum_{j=1}^{M}(1/Tr(YP^jY^T))^{1/(r-1)}}, j = \{1, \ldots, M\} \tag{35}$$

5. Experimental results

In this section, we experiment on both synthetic and real-world data sets to evaluate the performance of our method, named FLM. For LEM, LLE, HLLE, LTSA, and our Fusion of local manifolds (FLM) algorithms, we experiment on these data sets to obtain both visualization and quantitative evaluations. We utilize the global smoothness and co-directional consistence (GSCD) criteria [17] to quantitatively compare the embedding qualities of different algorithms: the smaller the value of GSCD, the higher the global smoothness, and the better the co-directional consistence. There are two adjustable parameters in our FLM method, that is, the tuning parameter r and the number of nearest neighbors k. FLM works well when the values of r and k are neither too small nor too large. The reason is that only one local method is chosen when r is too small, while the relative weights of different methods tend to be close to each other when it is too large. As a general recommendation, we suggest to work with $r \in [2, 6]$ and $k \in [0.7 \lceil log(N) \rceil, \, 2 \lceil log(N) \rceil]$.

5.1. Synthetic data sets

We first apply our FLM to the synthetic data sets that have been commonly used by other researchers: S-Curve, Swiss Hole, Punctured Sphere, and Toroidal Helix. The character of these data sets can be summarized as: general, non-convex, nonuniform, and noise, respectively. In each data set, we have total 1000 sample points, and the number of nearest neighbors is fixed to $k = 10$ for all the algorithms. For the S-Curve and Swiss Hole, we empirically set $r = 2$, and for the Punctured Sphere and Toroidal Helix data sets, we set r=3. **Figures 2–5** show the embedding results of the above algorithms on the four synthetic data sets. Each manifold learning algorithm and the corresponding GSCD result are shown in the title of each subplot. We can evaluate the performances of these methods by comparing the coloring of the data points, the smoothness, and the shape of the projection coordinates with their original mani-folds. **Figures 2–5** reveal the following interesting observations.

1. On some particular data sets, the traditional local manifold learning methods perform well. For example, LEM works well on the Toroidal Helix; LLE works well on the Punctured Sphere; HLLE works well on the S-Curve and Swiss Hole; and LTSA performs well on the S-Curve, Swiss Hole, and Punctured Sphere.

2. In general, our FLM performs the best on all the four data sets.

The above consequence is because only partial geometric information of the underlying man-ifold is learned by each traditional local manifold learning method, while the complementary geometric information learned from different manifold learning algorithms is respected by our FLM method.

5.2. Real-world data set

We next conduct experiments on the isometric feature mapping face (ISOFACE) data set [1], which contains 698 images of a 3-D human head. The ISOFACE data set is collected under different poses and lighting directions. The resolution of each image is 64×64. The intrinsic degrees of freedom are the horizontal rotation, vertical rotation, and lighting direction. The 2-D embedding results of different algorithms and the corresponding GSCD results are shown in **Figure 6**. In the embedding, we randomly mark about 8% points with red circles and attach their corresponding training images. In the experiment, we fix the number of nearest neighbors to $k = 12$ for all the algorithms. We empirically set r in FLM as 4. **Figure 6** reveals the following interesting observations.

1. As we can observe from **Figure 6b** and **c**, the embedding results of LEM and LLE show that the orientations of the faces change smoothly from left to right along the horizontal direc-tion, and the orientations of the faces change from down to up along the vertical direction. However, as we can see at the right-hand side of **Figure 6b** and **c**, the embedding results of both LEM and LLE come out to be severely compressed, and it is not obvious to survey the changes along the vertical direction.

2. As we can observe from **Figure 6d** and **e**, the horizontal rotation and variations in the brightness of the faces can be well revealed by the embedding result of HLLE and LTSA.
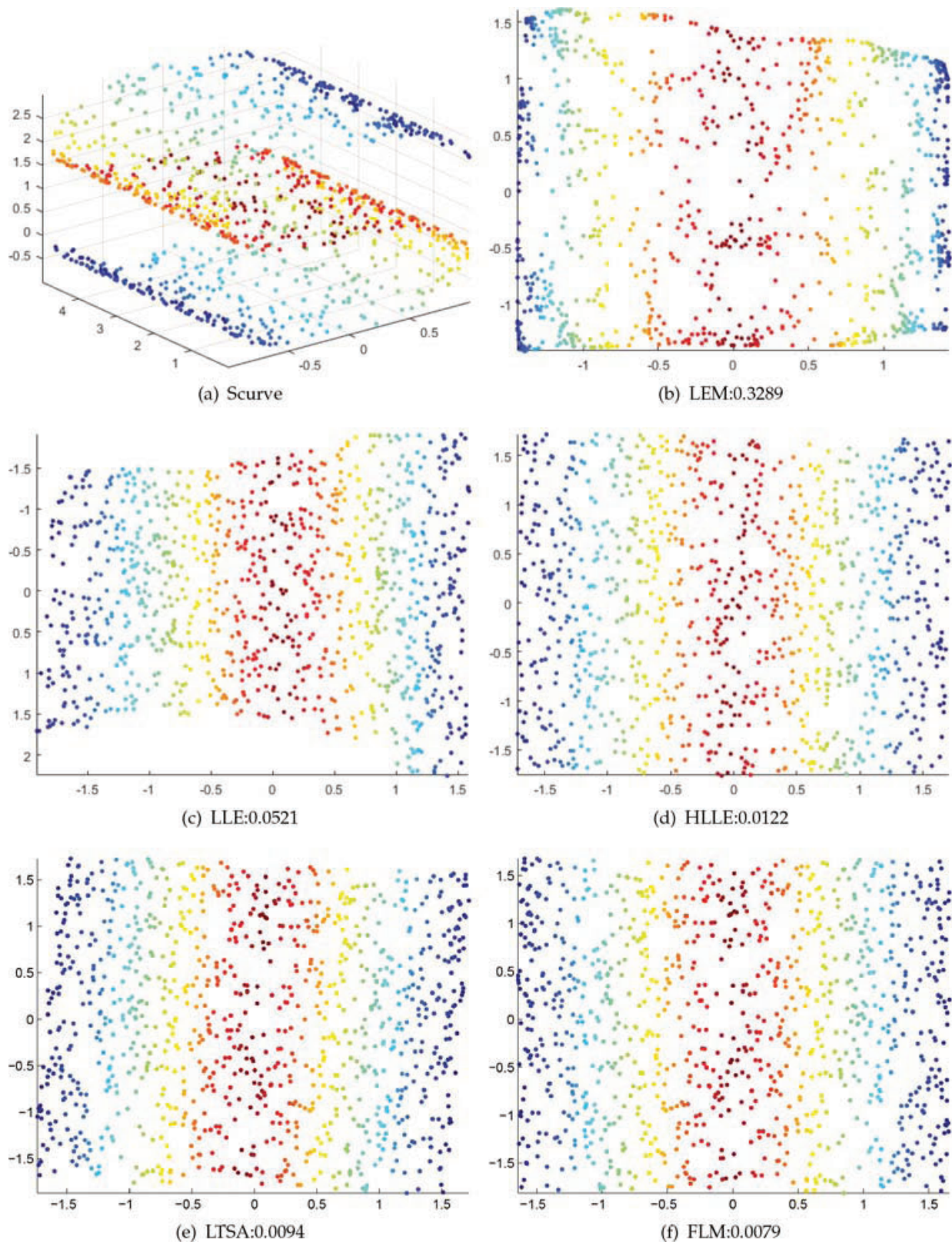
Figure 2. Embeddings of the synthetic manifold S-curve. The title of each subplot indicates the abbreviation of the manifold learning algorithm and the GSCD result. (a) Sample data. The title of subplots (b)-(f) indicates the abbreviation of the the manifold learning algorithm and the GSCD result.
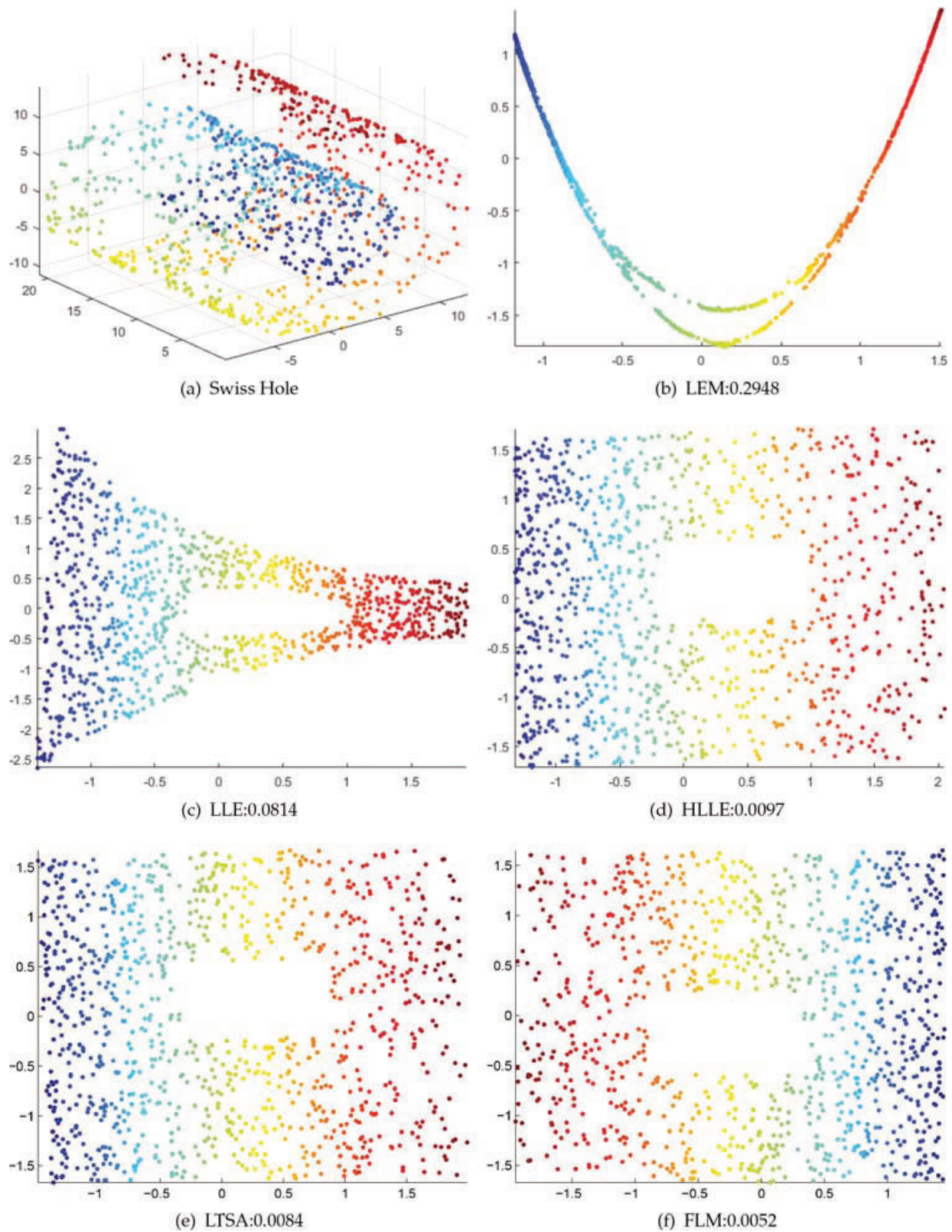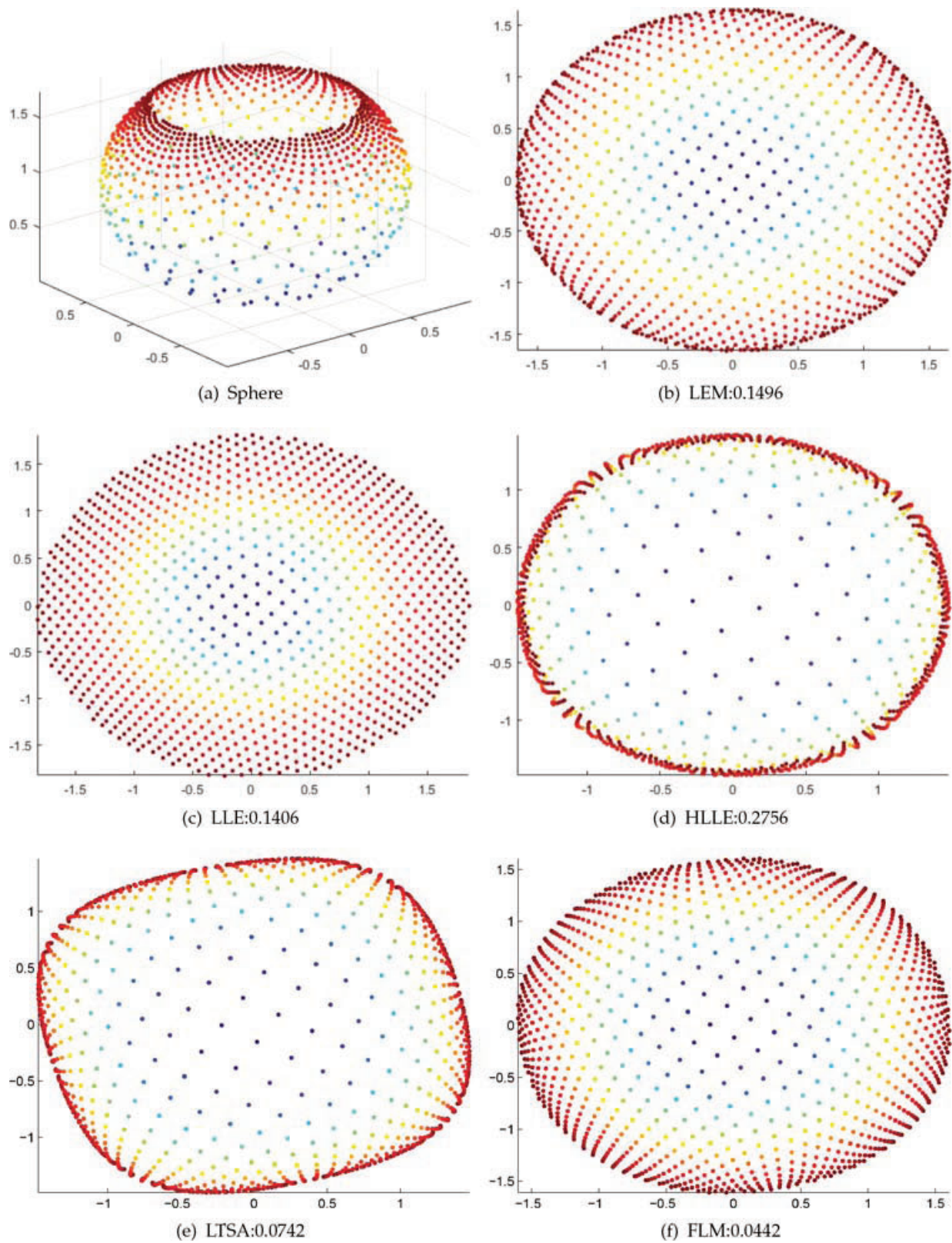
Figure 3. Embeddings of the synthetic manifolds Swiss Hole. The title of each subplot indicates the abbreviation of the manifold learning algorithm and the GSCD result. (a) Sample data. The title of subplots (b)-(f) indicates the abbreviation of the the manifold learning algorithm and the GSCD result.
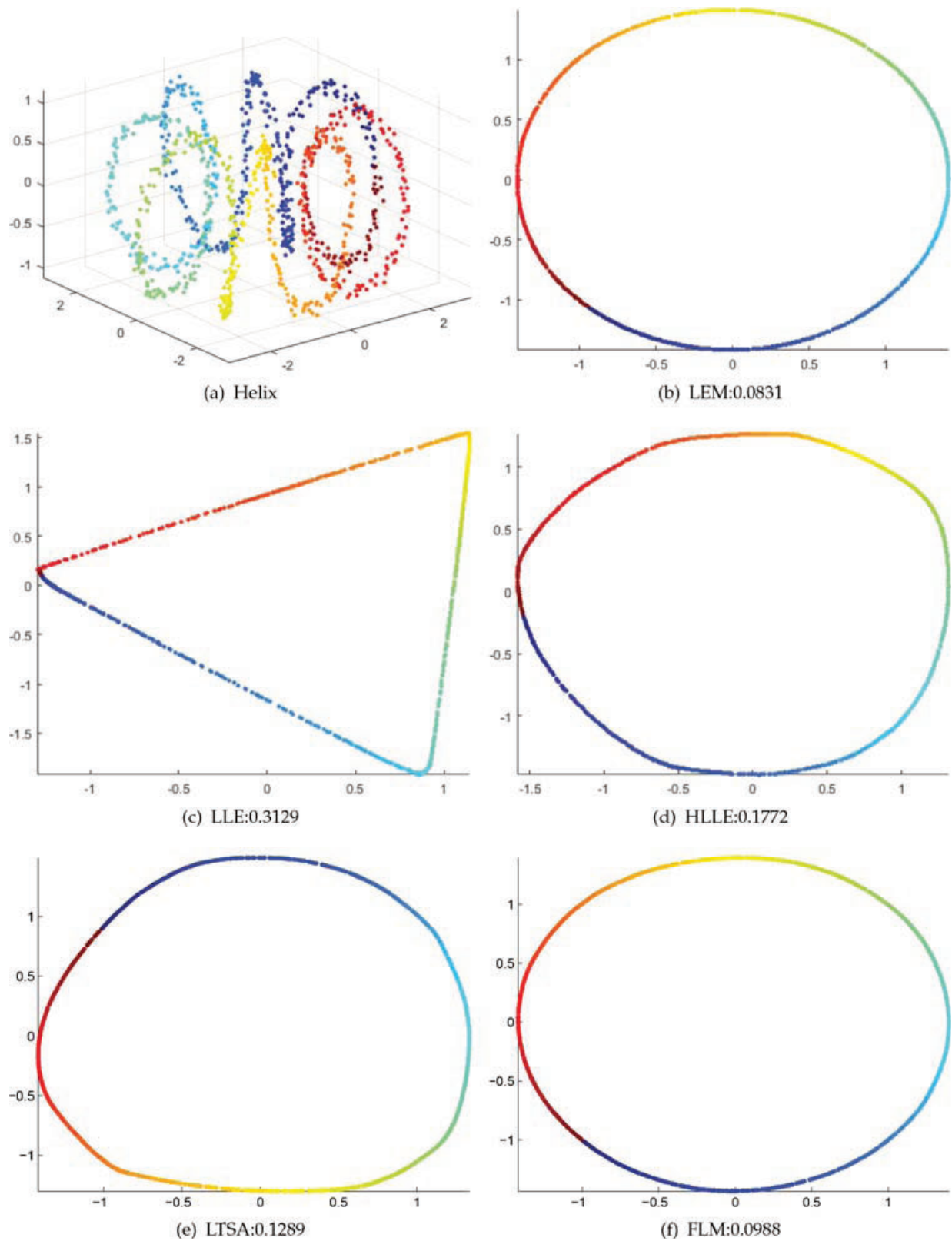
Figure 4. Embeddings of the synthetic manifolds Punctured Sphere. The title of each subplot indicates the abbreviation of the manifold learning algorithm and the GSCD result. (a) Sample data. The title of subplots (b)-(f) indicates the abbreviation of the the current manifold learning algorithm and the GSCD result.

Figure 5. Embeddings of the synthetic manifolds Toroidal Helix. The title of each subplot indicates the abbreviation of the manifold learning algorithm and the GSCD result. (a) Sample data. The title of subplots (b)-(f) indicates the abbreviation of the the manifold learning algorithm and the GSCD result.
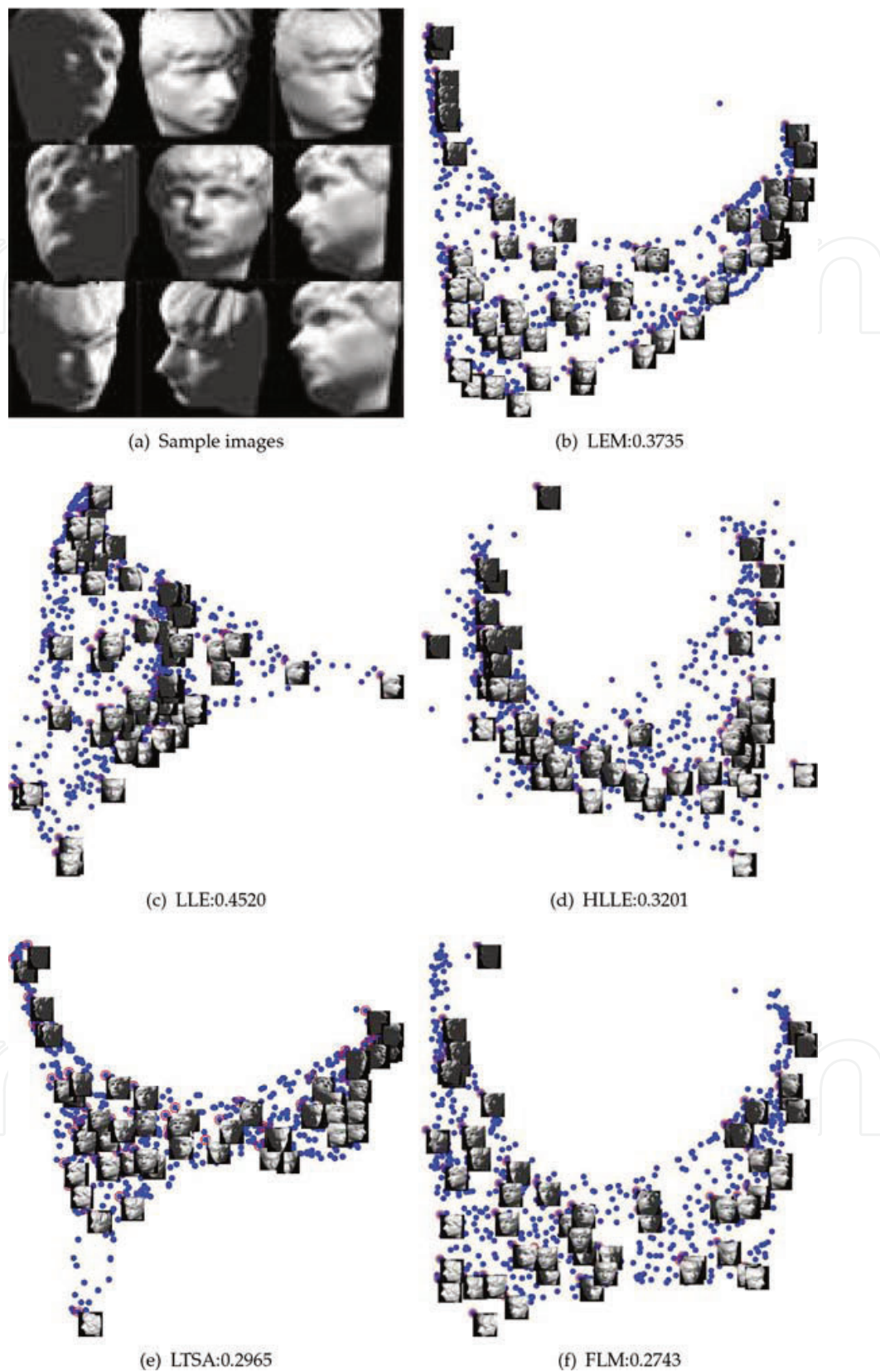
(a) Sample images

(b) LEM:0.3735

(c) LLE:0.4520

(d) HLLE:0.3201

(e) LTSA:0.2965

(f) FLM:0.2743

Figure 6. Embeddings of the ISOFACE data set. Subfigure (a) shows nine sample images, and subfigure (f) are the embedding results of different manifold learning algorithms. The title of each subplot indicates the abbreviation of the manifold learning algorithm and the GSCD result.

3. As we can observe from **Figure 6f**, orientations of the faces change smoothly from left to right along the horizontal direction, while the orientations of the faces change from down to up, and the light of the faces varies from bright to dark simultaneously along the vertical direction. These results illustrate that our FLM method successfully discovers the underlying manifold structure of the data set.

Our FLM performs the best on the ISOFACE data set, since our method makes full use of the complementary geometric information learned from different manifold learning methods. The corresponding GSCD results further verify the above visualization results in a quantitative way.

6. Conclusions

In this chapter, we introduce an interesting method, named FLM, which assumes a systematic framework to estimate the local objects and align them to reveal a single global low-dimensional coordinate space. Within the framework, we can fuse together the geometric information learned from different local methods easily and effectively to better discover the underlying manifold structure. Experimental results on both the synthetic and real-world data sets show that the proposed method leads to satisfactory results.

Acknowledgements

Author details

Xianglei Xing*, Kejun Wang and Weixing Feng

*Address all correspondence to: xingxl@hrbeu.edu.cn

College of Automation, Harbin Engineering University, China

References

[1] Tenenbaum JB, De Silva V, Langford JC. A global geometric framework for nonlinear dimensionality reduction. Science. 2000;290(5500):2319–2323.

[2] Weinberger KQ, Saul LK. Unsupervised learning of image manifolds by semidefinite programming. International Journal of Computer Vision. 2006;70(1):77–90.

[3] Chen J, Ma Z, Liu Y. Local coordinates alignment with global preservation for dimensionality reduction. IEEE Transactions on Neural Networks and Learning Systems. 2013;24(1):106–117.

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

[5] Roweis ST, Saul LK. Nonlinear dimensionality reduction by locally linear embedding. Science. 2000;290(5500):2323–2326.

[6] Donoho DL, Grimes C. Hessian eigenmaps: locally linear embedding techniques for high-dimensional data. Proceedings of the National Academy of Sciences. 2003;100(10):5591–5596.

[7] Zhang ZY, Zha HY. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. Journal of Shanghai University (English Edition). 2004;8 (4):406–424.

[8] Hou C, Wang J, Wu Y, Yi D. Local linear transformation embedding. Neurocomputing. 2009;72(10):2368–2378.

[9] Hou C, Zhang C, Wu Y, Jiao Y. Stable local dimensionality reduction approaches. Pattern Recognition. 2009;42(9):2054–2066.

[10] Wang R, Shan S, Chen X, Chen J, Gao W. Maximal linear embedding for dimensionality reduction. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2011;33 (9):1776–1792.

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

[12] Jolliffe I. Principal component analysis. Wiley Online Library; Springer-Verlag New York 2005.

[13] Turk M, Pentland A. Eigenfaces for recognition. Journal of Cognitive Neuroscience. 1991;3(1):71–86.

[14] Belkin M, Niyogi P. Towards a theoretical foundation for Laplacian-based manifold methods. Journal of Computer and System Sciences. 2008;74(8):1289–1308.

[15] Chern SS, Chen WH, Lam KS. Lectures on differential geometry. Vol. 1. World Scientific Pub Co Inc; Singapore 1999.

[16] James C. Bezdek, Richard J. Hathaway. Some notes on alternating optimization. In: Advances in Soft Computing. Springer; Springer Berlin Heidelberg 2002. p. 288–300.

[17] Zhang J, Wang Q, He L, Zhou ZH. Quantitative analysis of nonlinear embedding. IEEE Transactions on Neural Networks. 2011;22(12):1987–1998.