

Dimensionality Reduction for Local Coordinates Algorithm

S.M.Meena^{1,*}, V.S.Velladurai²P G Scholar, Department of ECE, Adithya institute of Technology, Coimbatore, India¹Assistant Professor, Department of ECE, Adithya institute of Technology, Coimbatore, India²

Abstract—Dimensionality reduction is vital in many fields, and alignment-based methods for nonlinear dimensionality reduction have become popular recently because they can map the highdimensional data into a low-dimensional subspace with the property of local isometry. However, the relationships between patches in original high-dimensional space cannot be ensured to be fully preserved during the alignment process. In this paper, we propose a novel method for nonlinear dimensionality reduction called local coordinates alignment with global preservation. We first introduce a reasonable definition of topology-preserving landmarks (TPLs), which not only contribute to preserving the global structure of datasets and constructing a collection of overlapping linear patches, but they also ensure that the right landmark is allocated to the new test point. Then, an existing method for dimensionality reduction that has good performance in preserving the global structure is used to derive the low-dimensional coordinates of TPLs. Local coordinates of each patch are derived using tangent space of the manifold at the corresponding landmark, and then these local coordinates are aligned into a global coordinate space with the set of landmarks in low-dimensional space as reference points. The proposed alignment method, called landmarks-based alignment, can produce a closed-form solution without any constraints, while most previous alignment-based methods impose the unit covariance constraint, which will result in the deficiency of global metrics and undesired rescaling of the manifold. Experiments on both synthetic and real-world datasets demonstrate the effectiveness of the proposed algorithm.

Index Terms—Isometric mapping, manifold learning, nonlinear dimensionality reduction, tangent space.

I. INTRODUCTION

THE PROBLEM of dimensionality reduction arises in many fields, such as machine learning, neural computation, data mining, and pattern recognition. The task of dimensionality reduction is to recover meaningful

low-dimensional structures hidden in high-dimensional data. An example might be a set of pixel images of an individual's face observed under different poses and lighting conditions; the task is to identify the underlying variables (pose angles, direction of light, etc.) given only the high-dimensional pixel image data. In many cases of interest, the observed data are found to lie on an embedded submanifold of the high-dimensional space. The degrees of freedom along this submanifold correspond to the underlying variables. In this form, the dimensionality reduction problem is known as manifold learning. Spectral methods have recently emerged as a powerful tool for dimensionality reduction and manifold learning. These methods are able to reveal low-dimensional structure in highdimensional data from the top or bottom eigenvectors of specially constructed matrices. To analyze data that lie on a low-dimensional submanifold, the matrices are constructed from sparse weighted graphs whose vertices represent input patterns and whose edges indicate neighborhood relations. The main computations for manifold learning are based on tractable polynomial-time optimizations, such as shortest path problems, least squares fit, semidefinite programming, and matrix diagonalization. Principal component analysis (PCA) [1] and metric multidimensional scaling (MDS) [2] are simple spectral methods for linear dimensionality reduction. Recently, from the viewpoint of manifold learning, some new linear methods have been proposed, such as locality preserving projection [3], neighborhood preserving embedding [4], local discriminant embedding [5], unsupervised discriminant projection [6], and orthogonal neighborhood preserving projections [7]. These methods are successfully only when the data manifold is linear. Recently, progress has been made in developing efficient algorithms to be able to learn the low-dimensional structure of nonlinear data manifolds. These proposed methods

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization,

Volume 3, Special Issue 1, February 2014

International Conference on Engineering Technology and Science-(ICETS'14)

On 10th & 11th February Organized by

Department of CIVIL, CSE, ECE, EEE, MECHANICAL Engg. and S&H of Muthayammal College of Engineering, Rasipuram, Tamilnadu, India

include isometric mapping (Isomap) [8], [9], locally linear embedding (LLE) [10], [11] and its variations [12]–[15],

Laplacian eigenmap (LE) [16], Hessian eigenmap [17], local tangent space alignment (LTSA) [18], maximum variance unfolding (MVU) [19], diffusion map [20], [21] and semisupervised generalized discriminant analysis [22]. The following

two strategies are shared in most of these algorithms:

1) exploiting the local geometry around each data point; and

2) mapping the manifold nonlinearly to a lower dimensional space based upon the learned local geometric information. Of course, these algorithms are different in the performance of local information extraction and global embedding. For example, LLE extracts the linearity by representing each point as a linear combination of its neighbors and then determines a low-dimensional embedding that preserves the locally linear combination structures. LLE is computationally efficient because it involves only sparse matrix computations which may yield a polynomial speedup; however, it neglects the global properties of the dataset. As a result, if two data points in the high-dimensional space are remote, LLE cannot ensure their corresponding data points in a lower dimensional space be remote. Some methods, such as Isomap, distance penalization embedding (DPE) [23], etc., try to keep both local and global properties; however, they have deficiencies to some extent. Isomap estimates the pairwise geodesic distance based on Euclidean distances between neighbors and then maps the high-dimensional points into a lower dimensional Euclidean space by preserving the geodesic distances. The derived geodesic distance matrix is inaccurate. In most cases, distances between neighbors are slightly smaller than the true value, and distances between far away data points are always larger than true. As a result, Isomap cannot derive local structure well. In addition, the mechanism of keeping the distances is effective only for globally isometric manifolds. For those manifolds that cannot be isometrically mapped to a lower dimensional Euclidean space, Isomap may fail. DPE tries to make the neighboring points to be still in the neighborhood and the remote points still to be remote by penalizing the distances. However, the penalization on local distances, as a looser constraint on neighborhood than constraints such as isometry and linear

reconstruction, may distort the true structure of the neighborhood. Sun *et al.* [23] proposed a two-step framework for nonlinear dimensionality reduction: in the first step, DPE is used to obtain the embedding of part of data points, and, in the second step, an additional constraint on the neighborhood relationships is imposed to refine local description by employing semisupervised manifold learning algorithms. However, the algorithm is very sensitive to parameters, and the parameter selections rely on some kind of performance evaluation of the dimensionality reduction algorithm, which is still an open problem. By combining the ideas of MVU and LE, Wang and Li [24], [25] proposed a manifold method that unfolds the dataset by maximizing the global variance subject to the proximity relation preservation constraint originated in LE. However, the method brings undesirable concentration on the boundary of the low-dimensional representation and its performance is very sensitive to parameters. Wang *et al.* [26] proposed locality-preserved maximum information projection (LPMIP) which considers both within locality and between locality in the processing of manifold learning. However, LPMIP is essentially linear and cannot provide reliable and robust solutions to nonlinearity distributions. Since Zhang *et al.* [18] proposed LTSA, which uses PCA to construct an approximation for the tangent space at each data point, and these tangent spaces are then aligned by a set of optimized affine transformations to give the global embedding, alignment technology has been a concern to many manifold learning researchers. The main advantage of alignment-based algorithms is that they can map the high-dimensional data into a low-dimensional subspace with the property of local isometry. LLE and LE are reformulated, respectively, using alignment technology in [27] and [28]. Three key issues in manifold learning determine the effectiveness of alignment-based algorithms. One issue is how to find a finite open cover of the manifold. A commonly used method is to constitute a neighborhood for each data point, and all of these neighborhoods naturally constitute a finite open cover of the manifold [18], [29]. However, the patches obtained this way are heavily overlapped and many of them can be ignored with little effect on manifold learning. To overcome it, Li [30] uses a greedy algorithm to combine a number of these patches into a larger patch. However, it is argued that the resulting patches will have the same fixed size and cannot be

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization,

Volume 3, Special Issue 1, February 2014

International Conference on Engineering Technology and Science-(ICETS'14)

On 10th & 11th February Organized by

Department of CIVIL, CSE, ECE, EEE, MECHANICAL Engg. and S&H of Muthayammal College of Engineering, Rasipuram, Tamilnadu, India

suitable to deal with the data nonuniformly sampled from a manifold [31]. In [32], a maximal linear patch (MLP) is first constructed for each data point under a geodesic distance constraint, and then a minimum subset of these MLPs is selected as a finite open cover of the manifold. The second issue is local isomorphic mapping. At present, PCA is often used to get the local coordinates of manifold [18], [30]. The last but not least issue is that of alignment, i.e., aligning the local coordinates of the manifold in the d -dimensional Euclidean space to form the global coordinate of the manifold. The most commonly used methods are the so-called global alignment methods. These methods align all local coordinates of the manifold, all simultaneously, by solving an eigenvalue problem [18], [27], [28], [30]. However, the large data size will lead to a large-scale eigenvalue problem which is hard to solve even for the state-of-the-art eigensolvers [33]. Moreover, the normalization constraints make them fail to preserve geodesic distances. Additionally, some progressive alignment methods [34], [35] select the local coordinate of a patch as the base local coordinate and then progressively expand the base local coordinate by aligning other local coordinates to the base local coordinate patch by patch until the base local coordinate becomes the global coordinate of the manifold. These progressive algorithms cannot ensure the minimum alignment error during each alignment, and the alignment errors will be accumulated and propagated during the process of progressive alignment. In this paper, we propose a new method for nonlinear dimensionality reduction, called local coordinates alignment with global preservation (LCA-GP). Compared with the existing methods, our method has the following features.

1) LCA-GP introduces a novel concept of topologypreserving landmarks (TPLs), which can preserve the global topological structure of manifold and construct a finite open cover of the manifold, i.e., a set of overlapped linear patches. The global nonlinear data structure is then represented by a collection of local coordinates as well as the set of TPLs.

2) LCA-GP learns the local manifold structures on a set of overlapping linear patches rather than on the neighborhood of each data point, which will reduce the redundancy.

3) LCA-GP aligns the local coordinates into a global low-dimensional coordinate space by landmarks-based alignment (LA), which can well preserve both the local geometry and the global structure of the manifold.

4) LCA-GP only needs to solve a linear system rather than an eigen problem with a unit covariance constraint as most spectral methods do. This kind of constraint will result in the deficiency of global metrics and undesired rescaling of the manifold.

5) LCA-GP approximately learns a nonlinear invertible mapping function in a closed form. Thus, the mapping can analytically project the test data points. This paper is organized as follows. Section II describes the motivation and basic ideas of the proposed LCA-GP. The detailed implementation of LCA-GP along with theoretical analysis is given in Section III. Section IV presents results of experiments on both synthetic and real-world datasets. Section V concludes this paper with directions for future work.

II. MOTIVATION AND BASIC IDEAS

Most spectral methods for nonlinear dimensionality reduction try to preserve the local geometry around each point

and solve optimization problems to obtain the global embedding. The global structure cannot be fully preserved by these methods due to their ignorance of the global properties of manifold. Moreover, the unit covariance constraint is imposed to obtain a unique solution, and the optimization problem turns into an eigenvalue problem. However, the imposed constraint should bring undesired relations among components of the embedding coordinates, which will result in the deficiency of global metrics and undesired rescaling of the manifold, as also pointed out in [31]. The question is how to preserve global structure as well as local geometry and avoid imposing the unit covariance constraint during solving the optimization problem. As described in the introduction, some works attempted to address all or part of these problems; however, each of them has some disadvantages. Inspired by the two-step framework for dimensionality reduction proposed by Sun *et al.* [23], we can select a set of representative points and use some global algorithm for dimensionality reduction to obtain lowdimensional coordinates of these points in the first step, and then develop some manifold learning algorithm, in which the optimization problem

under the constraint of a set of representative points can be transformed into a problem of solving linear system, to derive the embedding of all data points. There are two issues in the first step: one involves the selection of representative points and the other is to get the embedding coordinates of these selected points. The focus of this paper is the first issue. Representative data points, which we call landmarks, are selected from datasets that lie on an embedded manifold of the high-dimensional space. These landmarks should preserve the topology of the manifold and should be capable of learning the structure of certain curved manifolds. A criterion to test whether a landmark set can accurately and reliably represent the manifold was proposed in [36]. The criterion ensures that the individual Voronoi cells generated by the landmarks in the ambient space do not intersect the manifold at distant locations but it leaves out to ensure the linear structure of each cell. In [23], the centers of those neighborhoods that constitute a minimum set that cover the dataset are taken out as landmarks. The landmarks found in this way depend largely on the distribution density rather than some topological structure-relevant factors, such as curvature and so on.

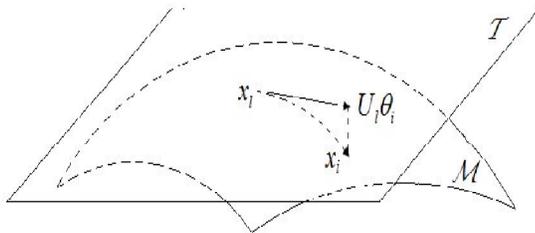


Fig. 1. Illustration of manifold M , tangent space T at a landmark x_l , and projection of the vector $x_i - x_l$ on T , $U_l \theta_i$, where θ_i is the coordinate of $U_l \theta_i$ under U_l , i.e., the local coordinate of a point x_i .

In this paper, we find a set of landmarks that not only ensures that for new test points the closest landmark in terms of Euclidean distance is also close in terms of geodesic distance on the manifold [36] but also can construct a set of corresponding neighborhoods such that these neighborhoods are linear and overlapped with each other and all together cover the whole dataset.

Specifically, for the latter, the approximation error of the point to the tangent space at the landmark is small enough. Consider the linear structure of a neighborhood of landmark $x_l = f(y_l)$, where $f: R^d \rightarrow R^D$, $d \ll D$, which can be characterized by the tangent space of the manifold at x_l .

A neighbor $x_i = f(y_i)$ of x_l can be represented by $x_i = x_l + J_f(y_l)(y_i - y_l) + \epsilon(y_i, y_l)$ (1) where $J_f(y_l)$ is the Jacobian matrix of f at y_l , whose columns span the tangent space, and $\epsilon(y_i, y_l)$ is a second order term of $y_i - y_l$, which measures the approximation error of x_i to the tangent space. Consider that the second-order term should be small, i.e., the first-order term should be close to $x_i - x_l$. With the assumption that the manifold is smooth enough and dataset is dense enough, we have

$$\|J_f(y_l)(y_i - y_l)\| \geq \eta \|x_i - x_l\| \quad (2)$$

where $\eta < 1$ is a parameter. If a set of orthogonal bases U_l of the tangent space of the manifold at each landmark,

where $U_l^T U_l = I$, have been attained and $U_l \theta_i \in T$ is a tangent vector [37], we have $\theta_i = U_l^T$

$U_l^T (x_i - x_l)$, where θ_i is the coordinate of $U_l \theta_i$ under U_l .

Then $U_l \theta_i$ is the estimation of $J_f(y_l)(y_i - y_l)$.

Evidently, $\|U_l \theta_i\| = \|\theta_i\|$. So we consider the following practical model:

$$\|\theta_i\| \geq \eta \|x_i - x_l\| \quad (3)$$

as a criterion to test whether a set of landmarks can reliably represent the manifold (see Fig. 1 for an illustration). Based on this criterion, in this paper we give a new concept of TPLs and propose an adaptive TPL selection method.

The second step is about the landmark-constrained manifold learning algorithm. As described in the introduction, three key issues in manifold learning determine the effectiveness of alignment based

algorithms. If there are a set of TPLs as well as their low-dimensional coordinates, landmarks-based methods on the three issues are proposed. At first, each data point is the neighbor of its two nearest landmarks, and the are approximately linear. Then tangent space of manifold at the landmark point, instead of at the mean point of patch, is used to derive the local coordinates of each patch. Finally, we take the low-dimensional coordinates of landmarks as constraints and transform the optimization problem in alignment into solving a linear system to obtain the last embedding. We call this LA. Briefly, the novelty of the proposed LCA-GP is twofold: the concept of TPLs and the LA method, which lead to several highlighted characteristics as described in the introduction part.

III. LCA-GP

In this section, we first introduce the concept of TPL, and the proposed method for TPL construction, as well as finite open cover construction. Then, the learning procedure of LCA-GP is presented in detail, including obtaining the global coordinates of TPLs, the construction of local model, and the landmarks-based alignment, i.e., the LA method. Finally, theoretical analyses of LCA-GP and comparisons of LCA-GP with other relevant methods are discussed.

A. TPLs and Finite Open Cover

Landmarks introduced in our algorithm not only contribute to preserving the global properties of the dataset and constructing a proper finite open cover of manifold but also ensure to allocate the right landmark to a new test point. The principal insight of the qualified set of landmarks lies in three criteria:

1) topological structure of manifold is well preserved by landmarks; 2) each data point is the neighbor of its two nearest landmarks, and the neighborhood of each landmark constitutes a patch, which spans a near linear subspace; and 3) for new test points, the closest landmark in terms of Euclidean distance is also close in terms of geodesic distance on the manifold. Those landmarks satisfying the above three criteria are called TPLs.

The first two criteria can consider that the approximation errors of each data point to the tangent space at its two nearest landmarks are small enough. As to the third one, we use the method proposed in [36]. Specifically, to each data point, the nearest landmark measured with

neighborhoods of all landmarks constitute the finite open cover of the manifold. We take each neighborhood as a patch. There should be common points in adjacent patches, and all patches Euclidean distance is the n th nearest one measured with geodesic distance, where $n \leq d+1$ and d is the intrinsic dimension of the manifold. Suppose $X = \{x_1, \dots, x_N\}$ is a set of N data points, where $x_t \in R^D, t = 1, \dots, N$, which are assumed to be drawn from a probability density, which is supported on a d -dimensional manifold M embedded in R^D , where $d < D$; and the landmarks will be chosen from X . Let the landmarks' indices be $L = l_1, l_2, \dots$. Then X_L is the set of landmarks, and x_{li} represents the i th landmark. A sketch of finding TPLs algorithm is given in Algorithm 1. The parameter α is the proportion of alternative initial landmark points. We set α as 0.05 without careful selection because it makes little difference to the result due to the following step of removing redundant landmarks. If the approximation error of one landmark x_{li} to the tangent space at one of the other landmarks is small, x_{li} is thought to be redundant and will be removed from the set of landmarks. The set T contains

Algorithm 1 Choose TPLs X_L -Sketch

Input: dataset $X = \{x_1, \dots, x_N\}$

Output: a set of TPLs $X_L = \{x_{l_1}, \dots, x_{l_p}\}$

1: initialization: $T = \emptyset, T_N = N$.

2: select $X_L \subset X$ randomly that $|X_L| = \alpha|X|$.

3: look for the redundant landmarks \hat{X}_L , and then $X_L \leftarrow X_L - \hat{X}_L$; compute L_E and L_M .

4: **while** $T_N \neq 0$ **do**

5: $T_N = 0, T = \emptyset$.

6: **for** each point $x_t \in X$ **do**

7: judge whether x_t is a TEP or not.

8: **if** x_t is a TEP **then**

9: $T = T \cup \{x_t\}; T_N = |T|$.

10: **end if**

11: **end for**

12: **if** $T_N \neq 0$ **then**

13: Apply Algorithm 2 to update X_L, L_E and L_M .

14: **end if**

15: **end while**

all of the topological error points (TEPs) corresponding to the current set of landmarks. The matrices L_E and L_M will be described in the next paragraph. The goal of the

algorithm is to find a set of landmarks. These landmarks can reliably represent the manifold, and can be used to construct linear patches. These landmarks also can generate individual Voronoi cells in the ambient space

including both x_t and $x_{L_j M(x_t)}$, $j = 1$ or 2 does not span a near linear subspace. However, the problem of how to define the criterion for enough on an unknown manifold is inherently ill-posed. According to the analysis in the above section and in [36], we thus turn to defining element of the vector LE in Algorithm 1 is $[LE(x_t)]$, and the t th column of the matrix LM is $[L_1 M(x_t) \dots L_{(d+1) M}(x_t)]^T$. We denote the inverse map for $L_j M(x_t)$ as $Cell_j M(l) = \{x_t | L_j M(x_t) = l\}$. The meaning of the "cell map" is as follows: given a landmark, find the data points that take this landmark as the j th landmark. TEP indicates that the projection error of x_t onto the tangent Euclidean distance in RD , and $dM(\cdot, \cdot)$ stand for the

that do not intersect the manifold at distant locations. If there are some TEPs, new landmarks should be added. Topological error points: let $dE(\cdot, \cdot)$ stand for the

geodesic distance in M (we derive the geodesic distance between two data

points using the algorithm in [8]). Given $x_t \in M$, $LE(x_t)$ denotes the index of its nearest landmark with respect to $dE(\cdot, \cdot)$, and $L_j M(x_t)$ denotes the index of its j th nearest landmark with respect to $dM(\cdot, \cdot)$ on M . The t th space of manifold at $x_{L_1 M(x_t)}$ or $x_{L_2 M(x_t)}$ is large, or x_t and $x_{LE(x_t)}$ are distant enough from each other such that the connection between them makes a short-circuit of that manifold. If only given the first criterion, $x_{L_1 M(x_t)}$ will be enough. Given the second criterion, as a result, both $x_{L_1 M(x_t)}$ and $x_{L_2 M(x_t)}$ are considered here. Then, the question is how to test whether x_t is a TEP. One needs to judge that whether the projection error of x_t onto the tangent space of manifold at $x_{L_1 M(x_t)}$ and $x_{L_2 M(x_t)}$ is large enough such that one patch heuristics.

1) Given a data point x_t , we find the number n such that $LE(x_t) = L_n M(x_t)$. Compute the x_t 's local coordinates

in the tangent space of manifold at $x_{L_j M(x_t)}$, $\theta_{j t}$, $j = 1, 2$. If $|\theta_{j t} - \theta_{x_t - x_{L_j M(x_t)}}| < \eta$, $j = 1$ or 2 , or $n > d+1$, x_t is taken as a TEP, where η is the parameter to determine the local linearity. Redundant landmarks can also be removed by this criterion (see step 3 in Algorithm 1). Specifically, if $|\theta_{L_{mn}} - \theta_{x_{lm} x_{ln}}| > \eta$, where $\theta_{L_{mn}}$ is x_{lm} 's local coordinate in the tangent space of manifold at x_{ln} , then x_{lm} is redundant and will be removed. We go through landmarks in a random order and remove the redundant ones sequentially. To eliminate the TEPs, new landmarks should be added. A simple solution is to choose a new landmark among those TEPs (step 13 in Algorithm 1). We first cluster TEPs, i.e., all $x \in T$, according to their $L_1 M(x)$. The second step is to choose a point in the biggest cluster $C(l_m)$ as the new landmark x_{l_q} by picking up the one that is the farthest from x_{l_m} . When a new landmark x_{l_q} is

Algorithm 2 Add New Landmark and Update L_E and L_M

- 1: cluster all $x \in T$ according to $L_M^1(x)$ and choose the biggest cluster $C(l_m)$
 - 2: $x_{l_q} \leftarrow \arg \max_{x \in C(l_m)} d_M(x, x_{l_m})$
 - 3: $X_L \leftarrow X_L \cup \{x_{l_q}\}$
 - 4: for each $x_t \in X$ do
 - 5: if $d_E(x_t, x_{l_q}) < d_E(x_t, x_{LE(x_t)})$ then
 - 6: $L_E(x_t) \leftarrow l_q$
 - 7: end if
 - 8: $\Delta = d_M(x_t, x_{L_M^{d+1}(x_t)})$
 - 9: if $d_M(x_t, x_{l_q}) < \Delta$ then
 - 10: find the minimum j , such that
 - 11: $d_M(x_t, x_{l_q}) < d_M(x_t, x_{L_M^j(x_t)})$
 - 12: $L_M^j(x_t) \leftarrow l_q$
 - 13: adjust $L_M^i(x_t)$, $i = j + 1, \dots, d + 1$, accordingly
 - 14: end if
 - 15: end for
-

acquired, we need to update $LE(x_t)$ and $L_{jM}(x_t)$, $j = 1, \dots, d + 1$ for all $x_t \in X$ (step 13 in Algorithm 1). The procedure is shown in Algorithm 2. Fig. 2 shows the unevenly sampled S-curve and the chosen landmarks for it by three methods, i.e., method in [36], method in [23], and our method. There are too few landmarks to preserve the global structure in Fig. 2(b). In Fig. 2(c), the landmarks are distributed unevenly as the origin data points do. There are too few landmarks to preserve the global structure in the sparse part and too many landmarks which will result in redundancy in dense part. In contrast, the TPLs found by our method can reflect the structure of manifold better [see Fig. 2 (d)]. Since the set of TPLs has been settled, the construction procedure for overlapping patches is straightforward. Suppose $XL = \{x_{l_1}, \dots, x_{l_p}\}$ is the set of TPLs. There are p overlapping linear patches corresponding to these landmarks. The i th patch

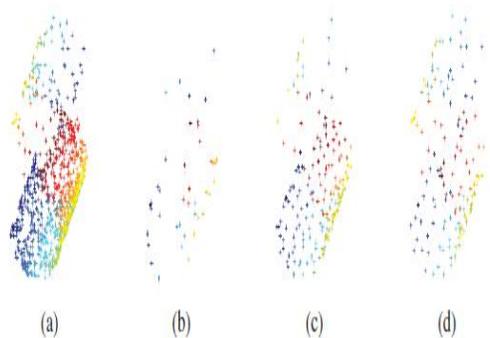


Fig. 2. Synthetic S-curve dataset. (a) 800 points sampled unevenly on S-curve manifold. (b) 41 landmarks chosen by method in [36]. (c) 173 landmarks chosen by method in [23]. (d) 138 TPLs. (For interpretation of the references to color in these figures, the reader is referred to the online version of this paper.)

$$X_i = \text{cell}_{\mathcal{M}}^1(l_i) \cup \text{cell}_{\mathcal{M}}^2(l_i). \quad (4)$$

C. Local Coordinates and Landmark-Based Alignment

B. Global Coordinates of TPLs

Some existing methods, such as MVU [19], Isomap [8], diffusion map [20], RML [31], and DPE [23], etc., which have good performance in preserving the global structure, can be used to derive the low-dimensional coordinates of TPLs. Because this point is not the focus of this paper and in view of the Isomap as a well-known classical algorithm, we chose an improved Isomap called topologically constrained isometric embedding (TCIE) proposed in [38] recently. All data points are used in the step of looking for the boundary points, and then TCIE is run on TPLs. We call it the global TCIE on landmarks.

Suppose $X = [x_1 \dots x_N]$ is the matrix of D -dimensional data, and $XL = [x_{l_1} \dots x_{l_p}]$ is the matrix of p landmark points which are the output of Algorithm 1. Global TCIE on landmarks (G-TCIE) has four steps.

1) Compute the $N \times N$ matrix of geodesic distances

$DM = dM(x_i, x_j)$. (In fact, this step has been done in Algorithm 1 in order to obtain LM .) Then, the geodesic distance matrix of landmarks is $DL = ST L DMSL$, where SL is a 0–1 data selection matrix such that $XL = XSL$.

2) Detect the boundary points ∂M of the data manifold. (Refer to [38].)

3) Detect a subset of consistent distances according to either (criterion 1) $\bar{P}_1 = \{(l_i, l_j) : cM(x_{l_i}, x_{l_j}) \cap \partial M = \emptyset\}$ where $cM(x_{l_i}, x_{l_j})$ is the geodesic connecting x_{l_i} and x_{l_j} , or (criterion 2, we use this criterion in our experiments) $\bar{P}_2 = \{(l_i, l_j) : dM(x_{l_i}, x_{l_j}) \leq dM(x_{l_i}, \partial M) + dM(x_{l_j}, \partial M)\}$ where $dM(x, \partial M) = \inf_{x' \in \partial M} dM(x, x')$ denotes the distance of x from the boundary of M .

4) Solve the MDS problem for consistent pairs only $Y * L = \text{argmin}_{Y_L} \sum_{l_i < l_j} w_{ij} (d_{l_i l_j}(Y_L) - dM(x_{l_i}, x_{l_j}))^2$

where $w_{ij} = 1$ if $(l_i, l_j) \in \bar{P}$ and $w_{ij} = 0$ otherwise.

By (4), we obtain p overlapping patches corresponding to p landmarks. We denote these patches as X_1, \dots, X_p , and the i th patch consists of a set of n_i data points in X . For a patch X_i with n_i data points, where the corresponding landmark is x_{li} and the matrix of the other $n_i - 1$ points is $\tilde{X}_i = [x_{i1} \dots x_{i(n_i-1)}]$, let $\tilde{X}_i = [x_{i1} - x_{li} \dots x_{i(n_i-1)} - x_{li}]$ be the neighborhood matrix of x_{li} . By SVD, we have

$$\begin{aligned} \tilde{X}_i &= U_i^{D \times r} \sigma_i^{r \times r} V_i^{r \times (n_i-1)} = U_i^{D \times r} \hat{\Theta}_i^{r \times (n_i-1)} \\ &= \begin{bmatrix} U_i^{D \times d} & U_i^{D \times (r-d)} \end{bmatrix} \begin{bmatrix} \hat{\Theta}_i^{d \times (n_i-1)} \\ \hat{\Theta}_i^{(r-d) \times (n_i-1)} \end{bmatrix} \\ &= U_i^{D \times d} \hat{\Theta}_i^{d \times (n_i-1)} + U_i^{D \times (r-d)} \hat{\Theta}_i^{(r-d) \times (n_i-1)}. \end{aligned} \quad (5)$$

Here, $d \ll D$, and the column vectors of $U_i^{D \times d}$ can be regarded as the orthonormal basis of the tangent space of manifold at x_{li} ; $U_i^{D \times d} \hat{\Theta}_i^{d \times (n_i-1)}$ is then the projection of \tilde{X}_i onto the tangent space. It is clear that $(U_i^{D \times d})^T \tilde{X}_i = (U_i^{D \times d})^T (U_i^{D \times d} \hat{\Theta}_i^{d \times (n_i-1)} + U_i^{D \times (r-d)} \hat{\Theta}_i^{(r-d) \times (n_i-1)}) = \hat{\Theta}_i^{d \times (n_i-1)}$. (6)

We find d largest eigenvalues $\lambda_1, \dots, \lambda_d$ and the corresponding eigenvectors $U_i = [u_{i1} \dots u_{id}]$ of X_i . The local low-dimensional coordinates of the data points denoted by $i = [\theta_{i1} \dots \theta_{i(n_i-1)}]$ in patch X_i are formulated as

$$\hat{\Theta}_i = U_i^T \tilde{X}_i. \quad (7)$$

It is easy to see that the local low-dimensional coordinate of x_{li} is a d -dimensional zero vector. Then we obtain the local coordinates of the whole patch i . Because the approximation errors of the data points in each patch to the tangent space at the corresponding landmark are small and these data points are thought to reside on a local linear space, the usage of the coordinates of data points on the tangent space of the manifold at landmark to obtain the local

representations becomes reasonable in our work. Let $Y = [y_1 \dots y_{N-p}]$ be the global coordinates of all input data points excluding p landmark points. i represents the local geometry of the manifold in patch X_i , and Y can be constructed by aligning all patches through translation, rotation, and scaling with landmarks as reference points, called LA. Formally, let Y_i represent the global coordinate of i in R^d . The low-dimensional coordinate of the i th landmark derived by TCIE is y_{li} , and let $y_{li} \in Y_i$, correspondingly, its local coordinate is θ_{li}, θ_{li}

$\in i$. As mentioned before, θ_{li} is a d -dimensional zero vector. Firstly, each patch is translated such that the corresponding landmark point θ_{li} meets y_{li} . In order to express easily, we can assume the i th landmark y_{li} to be the n_i th point in its patch Y_i . Taking out y_{li} from Y_i , we have $Y_i = [y_{i1} \dots y_{i(n_i-1)}]$. Subtracting y_{li} from all the data points in Y_i , we have where $\mathbf{1} = [1 \dots 1]^T$. Next we rotate and scale the patch to

$$\tilde{Y}_i = [y_{i1} - y_{li} \dots y_{i(n_i-1)} - y_{li}] = \hat{Y}_i - y_{li} \mathbf{1}^T \quad (8)$$

find the best A_i such that

$$\tilde{Y}_i \approx A_i \hat{\Theta}_i. \quad (9)$$

In (9), A_i represents a linear transformation matrix. The alignment error of local representations E_i can be expressed as

$$E_i = \tilde{Y}_i - A_i \hat{\Theta}_i. \quad (10)$$

In order to minimize the Frobenius norm $\|E_i\|_F$, we need to solve A_i using a least-squares fitting method. The solution is given by

$$A_i = \tilde{Y}_i \hat{\Theta}_i^\dagger \quad (11)$$

we can obtain

Let

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization,

Volume 3, Special Issue 1, February 2014

International Conference on Engineering Technology and Science-(ICETS'14)

On 10th & 11th February Organized by

Department of CIVIL, CSE, ECE, EEE, MECHANICAL Engg. and S&H of Muthayammal College of Engineering, Rasipuram, Tamilnadu, India

$$E_i = \tilde{Y}_i(I - \hat{\Theta}_i^\dagger \hat{\Theta}_i), \quad (12)$$

$$\hat{Y} = H_2 H_1^\dagger \quad (19)$$

$$W_i = I - \hat{\Theta}_i^\dagger \hat{\Theta}_i \quad (13)$$

where $H_{+1} = H_{T-1} (H_1 H_{T-1})^{-1}$ is the Moore–Penrose pseudoinverse of H_1 . Put Y_L and Y together, we get embedding of all data points Y .

and S_i be 0–1 data selection matrix such that $Y_i = Y S_i$. The squared alignment error of the data in patch X_i can be written as

$$\|E_i\|^2 = \|\tilde{Y}_i W_i\|^2 = \|(\hat{Y} S_i - y_i \Gamma^T) W_i\|^2, \quad (14)$$

From the global view, we need to seek a low-dimensional Y to minimize the total alignment squared error of all p patches. Thus, the final objective function of patches alignment can be derived as

$$\begin{aligned} E &= \sum_{i=1}^p \|E_i\|^2 \\ &= \left\| \left[(\hat{Y} S_1 - y_1 \Gamma^T) W_1 \cdots (\hat{Y} S_p - y_p \Gamma^T) W_p \right] \right\|^2 \\ &= \left\| \left[\hat{Y} S_1 W_1 \cdots \hat{Y} S_p W_p \right] \right. \\ &\quad \left. - \left[y_1 \Gamma^T W_1 \cdots y_p \Gamma^T W_p \right] \right\|^2 \\ &= \|\hat{Y} H_1 - H_2\|^2 \end{aligned} \quad (15)$$

where

$$H_1 = [S_1 W_1 \cdots S_p W_p] \quad (16)$$

$$H_2 = [y_1 \Gamma^T W_1 \cdots y_p \Gamma^T W_p]. \quad (17)$$

To minimize E in (15), we can find optimal Y by directly solving the linear system

$$\hat{Y} H_1 = H_2. \quad (18)$$

The last embedding can be expressed by

D. Proposed Algorithm and Approximate Analytic Projection

The main computation steps of LCA-GP can be summarized as follows.

- 1) Find a set of TPLs, X_L , by Algorithm 1.
- 2) Apply global TCIE on X_L and obtain the embedding Y_L in R_d .
- 3) Construct the overlapping patches set based on the landmarks by (4).
- 4) Obtain the local tangent coordinates of each patch X_i by (7) and compute W_i by (13).
- 5) Compute H_1 in (16) and H_2 in (17) and obtain the embedding Y by (19).

6) Put Y_L and Y together to get the last embedding Y . In the first step, computing LM and updating it involves calculating shortest paths from p TPLs to other nodes on the graph. With Fibonacci heap, these take $O(pN \log N)$.

Computing LE and updating it take $O(pN)$. As to run TCIE, since the geodesic distances between TPLs has been

calculated in the previous step, solving the MDS optimization problem takes $O(p^2 D)$ and the boundary detection takes $O(p^2)$. Calculating the local coordinates of all patches takes $O(N D \min(2N/p, D))$. The fifth step is to solve a linear system with the computational complexity $O(N^3)$ and for large dataset it consumes most of the time. Like almost all other methods for nonlinear dimensionality reduction, LCA-GP provides an embedding of a train set of

data points. Often, we need to project a test data point that is outside the train set. The problem is an “out-of-sample extension” [39]. In our algorithm, each data point belongs to two patches, patch i and patch j . We have

$$\begin{aligned}
 y_k &= A_i U_i^T (x_k - x_{l_i}) + y_{l_i} + e_i \\
 &= A_j U_j^T (x_k - x_{l_j}) + y_{l_j} + e_j \quad (20)
 \end{aligned}$$

where e_i and e_j are error terms and are very small. If the error term is ignored, then we approximately find transformation from each local linear model to the global coordinate space as follows:

$$\begin{aligned}
 y_{im} &= A_i U_i^T (x_{im} - x_{l_i}) + y_{l_i} \\
 m &= 1, \dots, n_i - 1, i = 1, \dots, p. \quad (21)
 \end{aligned}$$

The mapping function gives an explicit forward mapping from the observation space to the embedding space. Furthermore, its reverse mapping can be easily deduced in an entirely inverse manner

$$\begin{aligned}
 x_{im} &= U_i A_i^{-1} (y_{im} - y_{l_i}) + x_{l_i} \\
 m &= 1, \dots, n_i - 1, i = 1, \dots, p. \quad (22)
 \end{aligned}$$

Once the mapping function between the two spaces is built through a mixture of linear transformations, when applying to new test data, LCA-GP only needs first to identify to which patch the test data belongs and then to perform the corresponding transformation. Specifically, Algorithms 3 and 4 are designed to generalize the train results to unseen cases in the observation and embedding space, respectively. As a result, the train set is no longer required for subsequent process, leading to significant computational and storage savings.

Algorithm 3 Out-of-Sample Extension

Input: test data in the observation space $x_t \in R^D$

Output: d -dimensional embedding result $y_t \in R^d$

1: Assign x_t to the patch index j whose corresponding landmark is nearest to x_t :

$$j = \arg \min_i (\|x_t - x_{l_i}\|), i = 1, \dots, p$$

2: Compute the embedding coordinates:

$$y_t = A_j U_j^T (x_t - x_{l_j}) + y_{l_j}$$

Algorithm 4 Reconstruction Algorithm

Input: test data in the embedding space $y_t \in R^d$

Output: D -dimensional virtual example $x_t \in R^D$

1: Assign y_t to the patch index j whose corresponding landmark is nearest to y_t :

$$j = \arg \min_i (\|y_t - y_{l_i}\|), i = 1, \dots, p$$

2: Compute the reconstructed virtual example:

$$x_t = U_j A_j^{-1} (y_t - y_{l_j}) + x_{l_j}$$

E. Comparisons With Previous Work

It can be seen that LCA-GP bears some resemblance to LTSA and subsequent methods [30], [34]. Generally speaking, these methods all share the (similar) philosophy of aligning local coordinates in a global coordinate space.

However, there are some significant differences between LCA-GP and the other methods. First, the constituted finite

open covers of the manifold by these methods are different from each other. Second, LCA-GP solves the problem of optimum alignment differently from other methods. Progressive alignment method used in [34] and [35] is absolutely different from the proposed one. It aligns local coordinates patch by patch. Both LCAGP and the one in [18] and [30] try to minimize the global alignment error; however, as to minimize objective function, due to the introduction of landmarks, our LA algorithm is quite different from others which we call conventional alignment (CA). CA first translates each patch such that the local coordinate (which equals to zero) of the mean point of the patch meets its global coordinate (which is unknown), and then rotates and scales all patches to minimize the global alignment error as follows:

where S_i is selection matrix, W_i can be derived by local coordinates of the i th patch

$$\bar{E} = \sum_{i=1}^N \|Y S_i' W_i'\|^2 = \|Y B'\|^2 = \text{trace}(Y M' Y^T) \quad (23)$$

$$B' = (S_1' W_1' \cdots S_N' W_N') \quad (24)$$

and $M = B B^T$. Consider that the problem to minimize (23) is ill-posed, and a constraint, i.e., $Y Y^T = I$, is imposed to remove arbitrary translation and scaling. The LA method does not need to do so, and it first translates each patch such that the corresponding landmark's local coordinate (which equals to zero) meets its global coordinate (which is known), and then rotates and scales all patches to minimize the global alignment error as in (15) which can be translated into a linear system in (18) without any constraints. LCA-GP has properties similar to those of FusionGL [23]. First, both LCA-GP and FusionGL use landmarks to preserve the global structure

of manifold. Second, they fill the local models into the global coordinate space. We compare LCA-GP and FusionGL on these two issues. First, the two methods to select landmarks are different. It has been described in Section II. And the visualized comparison results are shown in Fig. 2. Second, FusionGL uses semisupervised manifold learning methods with inexact inputs, i.e., it minimizes an objective function that combines the mapping error of landmarks with a regularization term. Sun *et al.* use SS-LLE and mention that SS-LTSA can be used as well. We first compare LA with SS-LTSA. From alignment standpoint, SS-LTSA is same as LTSA on the first two steps, i.e., the neighborhood of each data point constitutes one patch and PCA is used to derive local coordinates of all patches. As to the third step, SS-LTSA translates each patch such that the local coordinate of the mean point of the patch meets its global coordinate, and then rotates and scales all patches such that the local coordinates of the landmarks meet their global coordinates, while the global alignment error is minimized, i.e., adds a constraint on (23). Then

$$\begin{aligned} \min_Y \quad & \text{trace}(Y M' Y^T) \\ \text{s.t.} \quad & Y S_L = Y_L \end{aligned} \quad (25)$$

where S_L is a selection matrix such that $X_L = X S_L$. From the semisupervised manifold learning standpoint, (15) in LA can be reformulated as follows:

$$\begin{aligned} E &= \left\| \left[(Y S_1 - Y R_1) W_1 \cdots (Y S_p - Y R_p) W_p \right] \right\|^2 \\ &= \|Y B\|^2 = \text{trace}(Y M Y^T) \end{aligned} \quad (26)$$

where S_i and W_i , $i = 1, \dots, p$, have the similar meaning of (14), R_i is 0-1 data selection matrix such that $Y R_i = y_{i-T}$

$$B = \left[(S_1 - R_1) W_1 \cdots (S_p - R_p) W_p \right] \quad (27)$$

and $M = BB^T$. Partitioning Y and M , minimization of (26) becomes

$$\min_{Y_1, Y_2} \text{tr} \left([Y_1 \ Y_2] \begin{bmatrix} M_{11} & M_{12} \\ M_{12}^T & M_{22} \end{bmatrix} \begin{bmatrix} Y_1^T \\ Y_2^T \end{bmatrix} \right) \quad (28)$$

where $Y_1 = Y_L$. By setting the gradient of the above objective function to zero, we get

$$Y_1 M_{12} + Y_2 M_{22} = 0. \quad (29)$$

We see that the global low-dimensional coordinates can be computed by solving a linear system of equations. Obviously, M in (26) is different from M_* in (25). For inexact landmarks information, FusionGL in fact minimizes

an objective function that combines the mapping error with a regularization term as follows:

$$\min_{Y_1, Y_2} \left([Y_1 \ Y_2] \begin{bmatrix} M'_{11} & M'_{12} \\ M'_{12} & M'_{22} \end{bmatrix} \begin{bmatrix} Y_1^T \\ Y_2^T \end{bmatrix} \right) + \beta(Y_1 - Y_1^*) \quad (30)$$

where $Y_1^* = Y_L$, β is the regularization parameter that reflects the confidence level in landmarks information. In this way, the global coordinates of landmarks can be optimized, but an approximate mapping function in closed form between high-dimensional data space and low-dimensional embedding space cannot be derived and the out-of-sample problem will not be settled.

IV. EXPERIMENTS

In this section, we systematically compare our method with other dimensionality reduction algorithms on both synthetic and real-world data. All algorithms are straightforwardly implemented in MATLAB.

In most experiments, our algorithm is compared with six different dimensionality-reduction algorithms based on manifold learning, i.e., Isomap [8], LLE [11], LTSA [18], LMDS [30], LLI [34], and FusionGL [23]. In these algorithms, Isomap, LLE, and LTSA are three well-known algorithms for nonlinear dimensionality reduction and LMDS and LLI are two representative alignment-

based algorithms devised recently. FusionGL is an algorithm similar to ours.

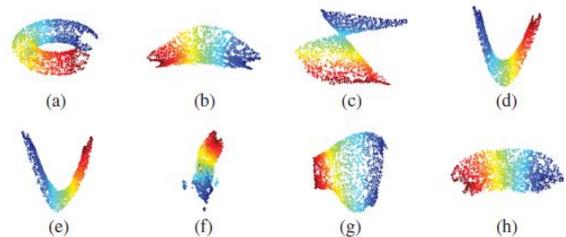


Fig. 3. (a) 2000 data points sampled from “incomplete tire” manifold. (b) Embedding by Isomap. (c) Embedding by LLE. (d) Embedding by LTSA. (e) Embedding by LMDS. (f) Embedding by LLI. (g) Embedding by FusionGL. (h) Embedding by LCA-GP. (For interpretation of the references to color in these figures, the reader is referred to the online version of this paper.)

A. Visualization Experiments

In order to test the effectiveness of LCA-GP in visualized way, two synthetic datasets, which are more difficult to learn than some widely used datasets such as Swissroll and Scurve, and one image dataset are first employed. The manifold of “incomplete tire” cannot isometrically map to a lower dimensional Euclidean space and the manifold of “Swissroll with a hole” (Swiss-hole) is nonconvex. The Frey face images dataset is also used in [11] and [30]. The corresponding 2-D embedding results are shown in Figs. 3–5, respectively. The first dataset contains 2000 data points evenly sampled from an “incomplete tire” and is plotted in Fig. 3(a). Seven algorithms run on all data points and the parameters of each algorithm are selected carefully as follows. The neighbor parameter k in Isomap, LLE, and LTSA is 6, 12, and 8, respectively. The number of centers in LLI is 150. In LMDS and FusionGL, the neighbor parameter k is 8 and the overlapping factor α is 0.5. The other three parameters of FusionGL are selected using the method in [23], specifically, $t_1 = 0.127$, $t_2 = \infty$, $\beta = 10^{-4}$. In LCA-GP, the proportion of initial landmarks parameter α is 0.05, the neighborhood size to estimate the tangent space of alternative landmarks is 6, and the local linearity parameter η is 0.985. One important future work is to develop an adaptive algorithm to automatically tune the parameter η by relying on some kind of performance

evaluation criterion [40]. The embedding result of Isomap contains many holes and distortion of shape [see Fig. 3(b)]. This is because Isomap cannot derive local structure well due to the inaccuracy of the computed geodesic distance matrix, and the mechanism of keeping the distances is only effective for globally iso

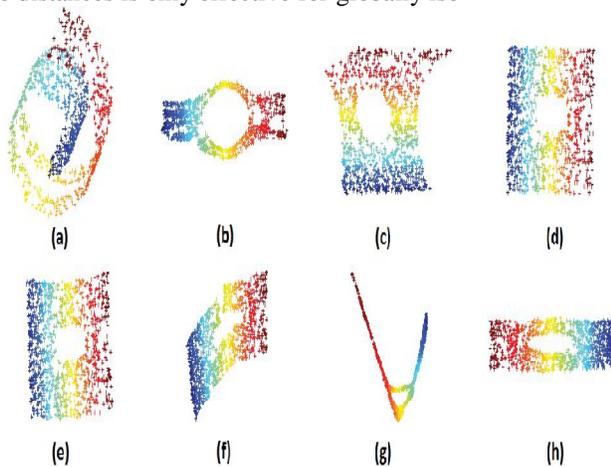


Fig. 4. (a) 1000 data points sampled from the Swiss-hole manifold. (b) Embedding by Isomap. (c) Embedding by LLE. (d) Embedding by LTSA. (e) Embedding by LMDS. (f) Embedding by LLI. (g) Embedding by FusionGL. (h) Embedding by LCA-GP. (For interpretation of the references to color in these figures, the reader is referred to the online version of this paper.)

TABLE I

COMPUTATION TIME(S) ON A PC WITH 2.4 GHz CPU AND 2G RAM,
USING MATLAB R2011B

| Datasets | Isomap | LLE | LTSA | LMDS |
|-----------------|---------|----------|--------|--------|
| Incomplete tire | 590.852 | 1.090 | 1.260 | 16.935 |
| Swiss-hole | 86.420 | 0.360 | 0.390 | 2.572 |
| Datasets | LLI | FusionGL | LCA-GP | |
| Incomplete tire | 143.250 | 145.977 | 32.190 | |
| Swiss-hole | 91.340 | 103.650 | 7.320 | |

neighborhood relations in the landmarks are wrong due to the existence of the hole and seriously distorted embedding of landmarks is derived by DPE based on these neighborhood relations. Our method [see Fig. 4(h)] finds of the manifold. This is due to the local isometric and global preserving properties of LCA-GP.

In Section III-D, the computational complexity of LCAGP is analyzed by Big O. To further compare with the

other algorithms, Table I reports the computation time of the experiments above. We see that LCA-GP is much faster than Isomap, LLI, and FusionGL, but slower than LLE, LTSA, and LMDS. Face recognition has been one of the most popular research topics in pattern recognition during this decade [41], [42]. A critical part in face recognition is the dimension reduction algorithm for feature extraction. We applied LCA-GP to the Frey face images dataset that has been used in [11] and [30]. The dataset consists of 1965 photographs taken of the same person with different poses and various facial expressions. Each photograph is a 20×28 grayscale image and therefore can be treated as a vector with 560 elements corresponding to raw pixel values, giving rise to inputs with $D = 560$ dimensions. Although the input dimensionality is quite high, the perceptually meaningful structure of these images is parameterized by fewer independent degrees of freedom pose and expression variables. We choose $\alpha = 0.05$,

$k = 16$, and $\eta = 0.6$ empirically. Fig. 5 illustrates the 2-D embedding and some reconstructions. Similar to previous work

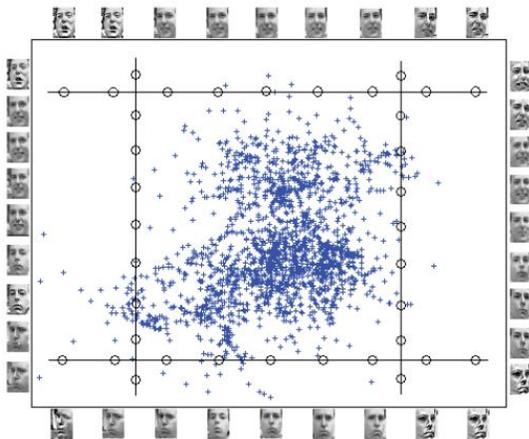


Fig. 5. 2-D embedding discovered by LCA-GP for the Frey face manifold.

Blue pluses indicate the global coordinate for each train example. The face images on the borders are reconstructed from the global coordinates specified by the corresponding open circles (data points along the straight lines in the global space).

[30], the 2-D embedding correctly discovers the two dominant variations in the face manifold, one for pose and another for expression. One may also see that some reconstructions, especially those near the boundary, are not good enough. This is mainly because the reconstruction algorithm is one kind of approximate algorithm, and, further, the model is extrapolating from the train images to low sample density regions.

B. Data Clustering

We apply the proposed and existing manifold learning algorithms to two handwritten digits databases, i.e., the USPS

dataset2 and the MNIST dataset,3 and then perform K -means clustering to compare the results of different algorithms in these tests. The USPS dataset contains 1100 grayscale images of each of the 10 digits from “0” to “9.” Each image is sized as 16×16 and converted to $D = 256$ dimensional vector. The MNIST dataset has a training set of 60 000 images and a testing set of 10 000 images. In our experiments, we take the first 1000 images from training set and the first 1000 images from testing set as our dataset. Each digit contains around 200 images. Each digit image is of size 28×28 and therefore represented by a $D = 784$ dimensional vector. The

clustering algorithm generates a cluster label for each data point. The clustering performance is evaluated by comparing the generated class label and the ground truth. In our experiment, accuracy is used to measure the clustering performance. Given a point x_i , let r_i and s_i be the obtained cluster label and the label provided by the ground truth, respectively. The accuracy is defined as follows:

$$\text{Accuracy} = \frac{\sum_{i=1}^N \delta(s_i, \text{map}(r_i))}{N} \quad (31)$$

TABLE II

CLUSTERING ACCURACY (PERCENT) BY USING THE 3-D EMBEDDING RESULTS ON THE USPS DATASET

| | Two Clusters | Three Clusters | Four Clusters | Average |
|----------|--------------|----------------|---------------|---------|
| Isomap | 71.56 | 63.38 | 60.88 | 65.27 |
| LLE | 84.19 | 80.25 | 80.17 | 81.54 |
| LTSA | 90.75 | 76.92 | 69.04 | 78.90 |
| LMDS | 89.12 | 76.42 | 68.83 | 78.12 |
| LLI | 82.56 | 80.58 | 78.68 | 80.60 |
| FusionGL | 88.87 | 78.25 | 76.91 | 81.34 |
| LCA-GP | 90.20 | 83.29 | 82.90 | 85.46 |

TABLE III

CLUSTERING ACCURACY (PERCENT) BY USING THE 2-D EMBEDDING RESULTS ON THE MNIST DATASET

| | Two Clusters | Three Clusters | Four Clusters | Average |
|----------|--------------|----------------|---------------|---------|
| Isomap | 62.73 | 59.20 | 56.64 | 59.52 |
| LLE | 77.58 | 75.71 | 75.26 | 76.18 |
| LTSA | 82.37 | 72.52 | 70.45 | 75.11 |
| LMDS | 81.92 | 71.31 | 69.64 | 74.29 |
| LLI | 74.65 | 73.81 | 71.68 | 73.38 |
| FusionGL | 78.79 | 72.58 | 70.14 | 73.84 |
| LCA-GP | 85.05 | 80.27 | 78.59 | 81.30 |

where N is the total number of data points and $\delta(x, y)$ is the delta function which is equal to 1 if $x = y$ and 0 otherwise, and $\text{map}(r_i)$ is the permutation mapping function that maps each cluster label r_i to the equivalent label from the dataset. The best mapping can be found by using the Kuhn–Munkres algorithm. The evaluations are conducted by using different numbers of clusters (K), i.e., $K = 2, 3, 4$. For each given cluster number K , 10 test runs are conducted on a subset of the data made of samples from K randomly chosen classes, and the final performance scores are computed by averaging the scores from the 10 tests. All algorithms are evaluated on the same combinations of K classes. We apply different algorithms to 3-D embedding (USPS) or 2-D embedding (MNIST) and apply K -means for clustering. The K -means is applied 10 times with different start points, and the best result in terms of the objective function of K -means is recorded. Table II shows the clustering accuracy by using 3-D embedding for each algorithm on the USPS dataset, and Table III shows the clustering accuracy by using 2-D embedding for each algorithm on the MNIST dataset. As can be seen, LCA-GP outperform the other six algorithms. In Table II, LTSA achieves 90.75% clustering accuracy while our method achieves 90.2% when the number of clusters is 2, but the average accuracy of LTSA is lower than ours. This is because the imposed unit covariance constraint of LTSA results in the deficiency of global metrics and undesired rescaling of the manifold, which may impact the clustering results, and the more the number of clusters, the larger the impact.

C. Pose Estimation

In this subsection, we use the oriental face database⁴ for head pose estimation to test the performance of LCA-GP.

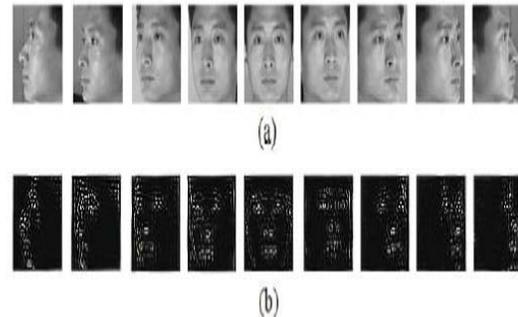


Fig. 6. (a) Face images for one subject. Only image corresponding to $(-80^\circ, \dots, 80^\circ)$ in increments of 20° are shown here. (b) Face images for the same subject shown in (a) which have been filtered with a derivative of Gaussian filter for edge enhancement.

The database consists of 33 669 images of 1247 people in JPEG format. We extract a typical subset of the oriental

face database, which contains face images of 20 different individuals; each individual has 19 face images with pose angle views varying from -90° to 90° in increments of 10° . The face samples are prepared in the following manner. First, face-only images are automatically extracted from the original images and further subjected to the following preprocessing: grayscale images are formed [see Fig. 6(a) for an example], which are subsequently transformed to edge images by using a derivative of Gaussian filter [see Fig. 6(b) for an example]. The resulting images are scaled to a constant size (20×21 pixels) and stored as 420-D vectors to be used as training/testing samples. Because the objective of the first experiment is to compare the performance of different manifold learning algorithms and most of the six algorithms listed at the beginning of Section IV do not address out-of sample extensions, we adopt the following experimental procedure: we run each of the seven methods (including ours) on all the 380 samples which

includes 200 training samples and 180 testing samples to derive the d -dimensional embedding of test samples and use a linear method proposed in [43] to obtain their view angles. Specifically, let X be all samples, X_m be the train

samples, and $X_{t \ i}$ be the test samples. For each manifold learning method, we first derive the d -dimensional embedding of X_m , Y_m , and a linear pose parameter map F is calculated from $m = FY_m$, where m is a matrix of the view angles of the training samples. Then the embedding of $X_{t \ i}$, $Y_{t \ i}$ is derived by running the corresponding manifold method on X . The resulting viewangles of the test samples can be estimated by $t \ i = FY_{t \ i}$.

Table IV illustrates the average angle errors obtained by each algorithm, as d is set to 2, 4, 6, 8, and 10, respectively.

The row corresponding to LCA-GP1 is the results of our method by the above procedure. As can be seen from Table IV, the proposed LCA-GP-based procedure finds more significant embedding space than the other six methods and achieves the lowest estimation errors. This may be due to the TPLs which can effectively guarantee the linear property in each local patch and to the LA method which can preserve the manifold structure better. To verify the effectiveness of out-of-sample extension of LCA-GP, we modify some steps of the above procedure. Specifically, $Y_{t \ i}$ is derived by running out-of-sample extension of LCA-GP. The row corresponding to LCA-GP2 in Table II shows the average angle errors obtained in this way. It can be

observation space and the embedding space are useful in real applications.

V. CONCLUSION

In this paper, we described a novel alignment-based manifold learning method. Compared with the classic Isomap and LLE, our algorithm can well preserve both the local geometry and global structure of the manifold. In comparison with the existing alignment-based methods such as LTSA and LMDS, LCA-GP derives more reasonable and efficient local models, and preserves the relationships between patches in original space during the alignment process without the unit covariance constraint. Compared with the related two-step method such as FusionGL, LCA-GP finds the TPLs which can reflect the structure of manifold better, and further approximately derives a parametric function for out-of-sample extension. Experimental results on both synthetic data and image datasets in Section IV indicate that LCA-GP obtains better embedding. Even so, still there are some problems that remain. For one thing, instead of empirical selection, developing an adaptive algorithm to automatically tune the parameter γ in LCA-GP

is a challenging and interesting task. Moreover, because of TCIE's failure for those manifolds that cannot be isometrically mapped to a lower dimensional Euclidean space, developing a more effective method to derive the global coordinates of landmarks may also be an important direction in future works. In addition, inspired by FusionGL, we propose to investigate a more flexible algorithm to optimize the global coordinates of landmarks in the LA step.

TABLE IV

AVERAGE ANGLE ERRORS (°) ACHIEVED BY EACH METHOD

| | $d = 2$ | $d = 4$ | $d = 6$ | $d = 8$ | $d = 10$ |
|----------|--------------|--------------|-------------|-------------|-------------|
| Isomap | 13.63 | 13.32 | 12.53 | 12.51 | 12.49 |
| LLE | 15.95 | 10.75 | 10.69 | 10.54 | 10.42 |
| LTSA | 16.49 | 14.67 | 13.52 | 13.23 | 12.12 |
| LMDS | 16.73 | 15.12 | 13.71 | 13.65 | 12.32 |
| LLI | 14.61 | 12.73 | 12.06 | 33.48 | 34.83 |
| FusionGL | 14.68 | 11.21 | 10.52 | 10.27 | 9.93 |
| LCA-GP1 | 11.05 | 10.34 | 9.82 | 9.43 | 9.27 |
| LCA-GP2 | 11.77 | 10.42 | 10.13 | 9.72 | 9.47 |

seen that, in all cases, the difference between LCA-GP1 and LCA-GP2 is very small. This indicates that the derived approximate analytic projections between the

REFERENCES

- [1] I. T. Jolliffe, *Principal Component Analysis*. New York: Springer-Verlag, 1999.
- [2] T. F. Cox and M. A. A. Cox, *Multidimensional Scaling*. London, U.K.: Chapman & Hall, 2001.
- [3] X. He, S. Yan, Y. Hu, P. Niyogi, and H. Zhang, "Face recognition using Laplacianfaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 3, pp. 328–340, Mar. 2005.
- [4] X. He, D. Cai, S. Yan, and H. Zhang, "Neighborhood preserving embedding," in *Proc. 10th IEEE Int. Conf. Comput. Vis.*, vol. 2, Mar. 2005, pp. 1208–1213.
- [5] H. Chen, H. Chang, and T. Liu, "Local discriminant embedding and its variants," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, vol. 2, 2005, pp. 846–853.

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization,

Volume 3, Special Issue 1, February 2014

International Conference on Engineering Technology and Science-(ICETS'14)On 10th & 11th February Organized by

Department of CIVIL, CSE, ECE, EEE, MECHANICAL Engg. and S&H of Muthayammal College of Engineering, Rasipuram, Tamilnadu, India

- [6] J. Yang, D. Zhang, J. Yang, and B. Niu, "Globally maximizing, locally minimizing: Unsupervised discriminant projection with applications to face and palm biometrics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 4, pp. 650–664, Apr. 2007.
- [7] E. Kokiopoulou and Y. Saad, "Orthogonal neighborhood preserving projections: A projection-based dimensionality reduction technique," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 12, pp. 2143–2156, Dec. 2007.
- [8] J. B. Tenenbaum, V. D. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimension reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.
- [9] M. Balasubramanian, E. L. Schwartz, J. B. Tenenbaum, V. De Silva, and J. C. Langford, "The isomap algorithm and topological stability," *Science*, vol. 295, no. 5552, p. 7, Jan. 2002.
- [10] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
- [11] L. K. Saul and S. T. Roweis, "Think globally, fit locally: Unsupervised learning of low dimensional manifolds," *J. Mach. Learn. Res.*, vol. 4, no. 2, pp. 119–155, 2004.
- [12] H. Chang and D.-Y. Yeung, "Robust locally linear embedding," *Pattern Recognit.*, vol. 39, no. 6, pp. 1053–1065, 2006.
- [13] X. Li, S. Lin, S. Yan, and D. Xu, "Discriminant locally linear embedding with high-order tensor data," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 2, pp. 342–352, Apr. 2008.
- [14] Y. Pan, S. S. Ge, and A. Al Mamun, "Weighted locally linear embedding for dimension reduction," *Pattern Recognit.*, vol. 42, no. 5, pp. 798–811, 2009.
- [15] C. Hou, J. Wang, Y. Wu, and D. Yi, "Local linear transformation embedding," *Neurocomputing*, vol. 72, nos. 10–12, pp. 2368–2378,

2009.

- [16] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Comput.*, vol. 15, no. 6, pp. 1373–1396, 2003.
- [17] D. Donoho and C. Grimes, "Hessian eigenmaps: New tools for nonlinear dimensionality reduction," *Proc. Nat. Acad. Sci.*, vol. 100, pp. 5591–5596, Mar. 2003.
- [18] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimensionality reduction via tangent space alignment," *SIAM J. Sci. Comput.*, vol. 26, no. 1, pp. 313–338, 2005.
- [19] K. Q. Weinberger and L. K. Saul, "Unsupervised learning of image manifolds by semidefinite programming," *Int. J. Comput. Vis.*, vol. 70, no. 1, pp. 77–90, 2006

BIOGRAPHY**MEENAL:**

BE (Electronics And Communication Engineering), Avinashilingam Institute For Home Science And Higher Education For Women University, Coimbatore , Passed Out In 2012. Me (Applied Electronics), Adithya Institute Of Technology, Coimbatore, Pursing 2014.