A Regularized Approach for Geodesic-Based Semisupervised Multimanifold Learning

Mingyu Fan, Xiaoqin Zhang, Zhouchen Lin, Zhongfei Zhang, and Hujun Bao

***Abstract*— Geodesic distance, as an essential measurement for data dissimilarity, has been successfully used in manifold learning. However, most geodesic distance-based manifold learning algorithms have two limitations when applied to classification: 1) class information is rarely used in computing the geodesic distances between data points on manifolds and 2) little attention has been paid to building an explicit dimension reduction mapping for extracting the discriminative information hidden in the geodesic distances. In this paper, we regard geodesic distance as a kind of kernel, which maps data from linearly inseparable space to linear separable distance space. In doing this, a new semisupervised manifold learning algorithm, namely regularized geodesic feature learning algorithm, is proposed. The method consists of three techniques: a semisupervised graph construction method, replacement of original data points with feature vectors which are built by geodesic distances, and a new semisupervised dimension reduction method for feature vectors. Experiments on the MNIST, USPS handwritten digit data sets, MIT CBCL face versus nonface data set, and an intelligent traffic data set show the effectiveness of the proposed algorithm.**

***Index Terms*— Feature extraction, manifold learning, semisupervised learning, image classification.**

I. Introduction

DIMENSION reduction plays an important role in classification problems, including face recognition [1], [2] and text clustering [3]. Classical linear dimension reduction methods, including Principal Component Analysis (PCA) [4], Linear Discriminant Analysis (LDA) [2], and Maximum Marginal Criterion (MMC) [5], are computationally efficient, globally optimal and in addition, converge asymptotically. Therefore, linear dimension reduction methods are commonly used in real applications. Recently, a number of studies show that image data possibly reside on nonlinear manifolds [6]–[8]. However, linear dimension reduction methods fail to discover the underlying nonlinear structure. As a promising approach to nonlinear dimension reduction, manifold learning becomes a hot topic and has been studied extensively.

Two promising manifold learning algorithms, Isometric Feature Mapping (Isomap) [6] and Locally Linear Embedding (LLE) [7], were introduced in the same issue of *SCIENCE* in 2000. Since then, many new manifold learning algorithms have been proposed based on different motivations, such as Laplacian Eigenmaps (LE) [8], Hessian LLE [9], and Local Tangent Space Alignment (LTSA) [10]. Manifold learning algorithms have an advantage over linear dimension reduction because they can extract the nonlinear structure of data. However, a common drawback of earlier manifold learning algorithms is that they learn the low-dimensional representations of high-dimensional data implicitly. No explicit mapping relationship from the input manifold to the output embedding can be obtained after the training process. Therefore, many linear projection based algorithms have been proposed for manifold learning by assuming that there exists a linear dimension reduction projection. Linear manifold learning algorithms include the Locally Preserving Projections (LPP) [11], Orthogonal Neighborhood Preserving Projections (ONPP) [12], Discriminative Orthogonal Neighborhood-Preserving Projections (DONPP) [13], and Graph embedding [14].

Geodesic distance, as an essential measurement for data dissimilarity, has been successfully used in manifold learning [6], [15], [16]. Roughly speaking, geodesic distance means the shortest path between points in the space. For Isomap [6] and related manifold learning methods [15], [16], geodesic distance is defined to be the sum of edge weights along the shortest path between two nodes (computed using Dijkstra's algorithm, for example). However, there are still limitations for the most of geodesic distance based manifold learning algorithms in classification. First, class information is rarely used in computing the geodesic distances between data points on manifolds. They are less effective when the dataset is partially labeled or distributes on multiple manifolds, as is common in classification. Second, little effort has been paid to build an explicit dimension reduction mapping for extracting the discriminative information hidden in geodesic distances.

In this paper, we consider geodesic distance as a kind of kernel, which maps data from linearly inseparable space to

Manuscript received July 27, 2013; revised December 21, 2013 and February 26, 2014; accepted March 6, 2014. Date of publication March 19, 2014; date of current version April 3, 2014. This work was supported in part by the NSFC under Grants 61203241, 61100147, 61272341, 61231002, 61305035, and 61121002, and in part by the NSF of Zhejiang Province under Grants LQ12F03004, LQ13F030009, and LY12F03016. The work of Z. Zhang was supported in part by the U.S. NSF under Grants IIS-0812114 and CCF-1017828, in part by the National Basic Research Program of China under Grant 2012CB316400, in part by the ZJU-Alibaba Joint Laboratory, and in part by the Zhejiang Engineering Center on Media Data Cloud Processing and Analysis. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Carlo S. Regazzoni.

M. Fan and X. Zhang are with the Institute of Intelligent System and Decision, Wenzhou University, Zhejiang 325035, China (e-mail: fanmingyu@amss.ac.cn; zhangxiaoqinnan@gmail.com).

Z. Lin is with the Key Laboratory of Machine Perception, School of Electrical and Engineering Computer Sciences, Peking University, Beijing 100871, China (e-mail: zlin@pku.edu.cn).

Z. Zhang is with the State University of New York, Binghamton, NY 13902 USA (e-mail: zhongfei@cs.binghamton.edu).

H. Bao is with the State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou 310058, China (e-mail: bao@cad.zju.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2014.2312643

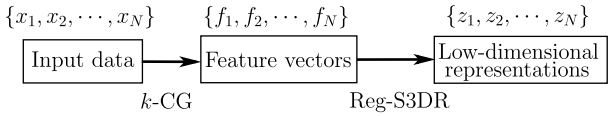


Fig. 1. The flow chart of the proposed Reg-GeoFeature algorithm.

linear separable feature space. In doing this, we propose a new semi-supervised manifold learning algorithm for image classification, which has three techniques. First, a semi-supervised neighborhood graph construction method is introduced for data distributed on multiple manifolds, which is named as k-Connectivity Graph (k-CG) method. Second, considering the geodesic distance as a kernel, we replace each data point with a distance feature vector, whose elements are graph distances from the data point to the remaining data points. This replacement is considered as a kind of feature mapping, which maps data from input space to linearly separable distance space. Third, to introduce the class information and to build explicit dimension reduction mappings, we propose a new semi-supervised linear dimension reduction method for the feature vectors. The new method, namely the Regularized Sparsity preserving Semi-Supervised Dimension Reduction (Reg-S3DR) method, maps the distance feature vectors from the same class to nearby locations and feature vectors from different classes to far away locations.

We combine the mapping to feature vectors and the linear dimension reduction method together to achieve the explicit nonlinear dimension reduction mapping. The flow chart of our Regularized Geodesic Feature Learning (Reg-GeoFeature) algorithm is given in Fig. 1, where $\{x_1, \ldots, x_N\}$ denotes the input dataset, f_i is the corresponding feature vector, which is used to replace x_i for $i = 1, \ldots, N$, and $\{z_1, \ldots, z_N\}$ are the obtained low-dimensional representations of $\{f_1, \ldots, f_N\}$.

The research herein extends and improves upon the research of [17], through the following.

1. Theoretical results on the validity and the rationality of the replacement of original data points to geodesic distance feature vectors are included.
2. More related state-of-the-art supervised and semi-supervised manifold learning methods [18]–[20] are compared with our method for image classification.
3. We have conducted experiments on another two benchmark datasets, namely the intelligent traffic dataset [45] and the MNIST handwritten dataset [47].
4. In order to better use the label information and manifold structure of data for semi-supervised learning, we replace the Semi-Supervised Discriminant Analysis method [17] to a more robust semi-supervised linear dimension reduction method, which is called as the Regularized Sparsity preserving Semi-Supervised Dimension (Reg-S3DR).

The rest of the paper is structured as follows. In Section II, the related dimension reduction algorithms are reviewed. Our algorithm is described in Section III. In Section IV, experiments are conducted on real world datasets to show the promise and effectiveness of the proposed Reg-GeoFeature algorithm. Finally, in Section V, we provide concluding remarks and suggestions for the future work.

TABLE I
NOTATION

$\mathbb{R}^D$	The input space, D-dimensional Euclidean space.
$\mathbb{R}^d$	The output space, d-dimensional Euclidean space.
$\mathcal{X}$	$\mathcal{X} = \{x_1, \cdots, x_l, \cdots, x_{u+l}\}$ with $x_i \in \mathbb{R}^D$, the total dataset. $\{x_i\}_{i=1}^{l}$ are labeled points, and $\{x_i\}_{i=l+1}^{u+l}$ are unlabeled points.
N	$N = u + l$, the total number of data points in $\mathcal{X}$.
X	$X = [x_1, \cdots, x_N] \in \mathbb{R}^{D \times N}$ is the input data matrix.
Z	$Z = [z_1, z_2, \cdots, z_N] \in \mathbb{R}^{d \times N}$ is the output matrix. $z_i \in \mathbb{R}^d$ denotes the low-dimensional representation of data point f_i.
$\mathcal{Y}$	$\mathcal{Y} = \{y_1, \cdots, y_l\}$ is the label set. $y_i \in \{1, 2, \cdots, C\}$ is the label of data point x_i.
$\mathcal{F}$	$\mathcal{F} = \{f_1, \cdots, f_N\}$. $f_i = (d_G(x_i, x_1), \cdots, d_G(x_i, x_N))^T$ is the feature vector of x_i. If x_i is labeled as y_i, f_i is labeled as y_i.
$\mathcal{X}_m$	$\mathcal{X}_m = \{x_{m,1}, \cdots, x_{m,N_m}\}$ with $x_{m,j} \in \mathcal{X}$. Data points of the m-th class, for $m = 1, \cdots, C$.
C	The number of classes that the data points belong to.
N_m	The number of data points in the m-th class.
$\mathcal{S}^p$	$\mathcal{S}^p = \{x_1^{(p)}, \cdots, x_{n_p}^{(p)}\}$ with $x_j^{(p)} \in \mathcal{X}$. Data points of the p-th manifold, for $p = 1, \cdots, P$.
$x_{q(i)}^p$	$(x_{q(i)}^p, x_{p(i)}^q)$ is the i-th shortest edge between data manifolds $\mathcal{S}^p$ and $\mathcal{S}^q$. $x_{q(i)}^p$ denotes the ending vertex from $\mathcal{S}^p$.

II. Related Work

In order to avoid confusion, we give a list of the main notations used in this paper, as shown in Table I.

There are a lot of successful supervised dimension reduction algorithms. LDA [2] is designed to find a linear projection A which maximizes the distances among the means of the classes and minimizes the distances among the points in the same class using the Fisher's criterion:

$$A^* = \arg\max_{A \in \mathbb{R}^{N \times d}} \frac{\mathrm{tr}\left(A^T S_b A\right)}{\mathrm{tr}\left(A^T S_w A\right)}, \tag{1}$$

where the within-class scatter matrix S_w and the between-class scatter matrix S_b are defined as

$$S_w = \sum_{m=1}^{C} \sum_{j=1}^{N_m} (x_{m,j} - c_m)(x_{m,j} - c_m)^T, \tag{2}$$

$$S_b = \sum_{m=1}^{C} N_m (c_m - c)(c_m - c)^T, \tag{3}$$

with $c_m = \frac{1}{N_m} \sum_{j=1}^{N_m} x_{m,j}$ as the mean of data points in the m-th class and $c = \frac{1}{N} \sum_{i=1}^{N} x_i$ as the mean of all data points. Despite the success of LDA [2], it has been found to have intrinsic problems [21]: singularity of within-class scatter matrices and limited available projection directions. Many subspace based variants of LDA have been done to deal with these problems, such as [22]–[24].

MMC [5] is based on the same intuition as LDA. The algorithm takes the following approach to find a dimension reduction mapping:

$$A^* = \arg\max_{A \in \mathbb{R}^{N \times d}, A^T A = I} \mathrm{tr}\left(A^T (S_b - S_w) A\right). \tag{4}$$

Cai et al. [25] proposed a semi-supervised discriminant analysis method, which includes a term which preserves the

local information of the unlabeled data points to improve the performance of the classification. The term is

$$J(A) = \frac{1}{2}\sum_{i,j} \|A^T x_i - A^T x_j\|^2 w_{ij} = \mathrm{tr}\left(A^T X L X^T A\right), \quad (5)$$

where $X = [x_1, \ldots, x_N]$ is the data matrix, $L = D - W$ be the graph Laplacian matrix, $W = \left(w_{ij}\right)$ is the similarity matrix of order N, and D is the $N \times N$ diagonal matrix with $D_{ii} = \sum_j w_{ij}$ for $i = 1, \ldots, N$. The objective function for the algorithm is presented as

$$A^* = \arg\max_{A \in \mathbb{R}^{N \times d}} \frac{\mathrm{tr}\left(A^T S_b A\right)}{\mathrm{tr}\left(A^T (S_t + \alpha X L X^T) A\right)}, \quad (6)$$

where $S_t = S_b + S_w$ and $\alpha > 0$ is a given parameter. Kernelized Semi-Supervised Discriminant Analysis [25] is also proposed to discover the nonlinear intrinsic geometry. Independently, Song et al. [26] proposed a similar semi-supervised dimension reduction framework based on the LDA and the MMC algorithms. In [27], Nie et al. proposed a flexible manifold embedding framework, which unifies a lot of graph embedding related, linear, and kernelized nonlinear semi-supervised dimension reduction methods. All of these semi-supervised dimension reduction methods work by adding a manifold smooth term to an optimization problem. Because these methods construct the adjacent graphs in an unsupervised manner, class information of the partially labeled data is not well used in discovering the discriminant structure of the data manifold.

The Discriminative Multi-Manifold Analysis (DMMA) [33] for face recognition first segments each face image into non-overlapping patches in a specific way and then treats all the patches of an image as a data manifold. Discriminant Analysis then is implemented on the data manifolds for feature extraction. In the assumption of DMMA, there is high overlapping between these manifolds and the distances between patches at the same location of different images are usually smaller than those at different locations of the same image. For example, the similarity of the patches of two eyes from two different subjects is usually higher than that of an eye and a cheek from the same person. However, this assumption is no longer valid for non-face image classification problems because the patches from the same location of different images (from different classes) may be highly dissimilar. Compressive sensing is also a powerful tool in subspace learning. There are multi-subspace learning algorithms based on the sparse representation [34] or low-rank representation [35]. But previous studies indicate that sparsity based methods are usually computationally expensive [34], [36].

In order to incorporate class information into manifold learning, the S-Isomap method [37] is proposed. This algorithm replaces the Euclidean distance with a new measurement of dissimilarity between data points: $D(x_i, x_j) = \sqrt{1 - e^{-\|x_i - x_j\|/\beta}}$ if $y_i = y_j$ and $D(x_i, x_j) = \sqrt{e^{\|x_i - x_j\|/\beta}} - \alpha$ if $y_i \neq y_j$, where α and β are pre-specified parameters, y_i is the class label of x_i. The subsequent procedure of S-Isomap algorithm is the same as that of Isomap algorithm. However, a general method to determine the appropriate values of parameters α and β is still unknown. E-Isomap [38] is another supervised manifold learning algorithm which has three steps. The first and second steps of E-Isomap are the same as those of the classical Isomap algorithm. In the third step, a feature vector f_i is used to represent the original data point x_i, where $f_i = (d_G(x_i, x_1), \ldots, d_G(x_i, x_N))^T$, for $i = 1, \ldots, N$, where $d_G(x_i, x_j)$ denotes the graph distances between data points on the adjacent graphs. The classical LDA [2] method is then applied to reduce the dimension of the extracted feature vectors $\{f_1, \ldots, f_N\}$. Therefore, E-Isomap suffers the intrinsic problem of LDA: limited available projection dimensions.

By introducing the local metrics of semi-Riemannian manifold to describe the structures of classes, Zhao et al. [31] proposed the Semi-Riemannian Discriminant Analysis (SRDA) algorithm for supervised dimension reduction. An extended SRDA algorithm for semi-supervised dimension reduction has been proposed in [32]. Isometric Projection (IsoProjection) [18] constructs a weighted data graph with the approximations of the geodesic distances. A linear projection mapping is then obtained by preserving the pair wise distances in the graph embedding manner. In [19], [20], Maximum Margin Projection (MMP) and Locality Sensitive Discriminant Analysis (LSDA) are both designed for discovering the local manifold structure in the semi-supervised manner. There are also several related multi-manifold analysis methods, such as [28]–[30]. However, the projective mappings proposed in these methods either work on the original data points or on the kernel vectors. None of them explores the discriminative information hidden in geodesic distances.

III. A Regularized Approach for Semi-Supervised Multi-Manifold Learning

In this section, we propose the Reg-GeoFeature algorithm with the following three features for multi-manifold learning.

1. A new neighborhood graph construction method is proposed and used in our algorithm. This feature is presented in Section III-A.
2. Each data point is replaced with a feature vector built by graph distances from this point to the remaining data points. This feature is presented in Section III-A.
3. A regularized sparsity preserving dimension reduction method is proposed to build explicit mapping from feature vectors to low-dimensional representations. This feature is also presented in Section III-B.

A. Multi-Manifold Modeling and Distance Feature Vectors Building

In this subsection, we consider the construction of a neighborhood graph for multi-manifold data and the replacement of the original data points with the feature vectors built from the graph distances.

That data lying on multiple manifolds are common in the real world. For instance, in face recognition the images of each person form his or her own manifold [39]; in computer vision motion detection and human tracking, moving subjects trace different trajectories which are low-dimensional

Algorithm 1 The k-CG Algorithm

1: Input: Euclidean distance matrix D, whose (i,j)-th entry is $\|x_i - x_j\|$, neighborhood size k or ε.
2: Output: Graph $G = (\mathcal{X}, \mathcal{E})$ ($\mathcal{X}$ is the set of vertices and $\mathcal{E}$ is the set of edges in the graph), the number P of graph partitions.
3: Initialization: $\mathcal{V} = \{x_1, \cdots, x_N\}$, $\mathcal{E} = Queue = \emptyset$, $tempC = 1$
4: **for** $i = 1$ to N **do**
5: Identify the nearest neighbors $\{x_{i1}, \cdots, x_{il_i}\}$ for x_i by k-nearest neighbor or ε-nearest neighbor in a semi-supervised manner. Let

$$\mathcal{E} = \mathcal{E} \bigcup \{(x_i, x_{i1}), \cdots, (x_i, x_{il_i})\} \tag{7}$$

6: **end for**
7: **while** $\mathcal{V}$ is not empty **do**
8: Take out any data point x from $\mathcal{V}$ and assign its graph partition:

$$\begin{aligned} Queue &\leftarrow Queue + \{x\}, \quad label(x) = tempC, \\ \mathcal{V} &\leftarrow \mathcal{V} - \{x\} \end{aligned} \tag{8}$$

9: **while** $Queue$ is not empty **do**
10: Extract x from $Queue$: $Queue \leftarrow Queue - \{x\}$
11: $\forall \tilde{x}$: $\tilde{x}$ is a nearest-neighbor of x
12: **if** $\tilde{x}$ is not labeled **then**
13: Assign $\tilde{x}$ to the same graph partition with x:

$$\begin{aligned} Queue &\leftarrow Queue + \{\tilde{x}\}, \quad label(\tilde{x}) = tempC, \\ \mathcal{V} &\leftarrow \mathcal{V} - \{\tilde{x}\} \end{aligned} \tag{9}$$

14: **end if**
15: **end while**
16: $tempC = tempC + 1$
17: **end while**
18: $P = tempC - 1$
19: **if** $P \geq 2$ **then**
20: Let l_i be the number of neighbors associated with x_i and compute the average number of nearest neighbors: $k = \frac{1}{N}(\sum_{i=1}^{N} l_i)$
21: **end if**
22: **for** $p = 1$ to P **do**
23: **for** $q = p + 1$ to P **do**
24: Identify the k shortest inter-partition edges between $\mathcal{S}^p$ and $\mathcal{S}^q$, which are referred to as $\{(x^p_{q(1)}, x^q_{p(1)}), \cdots, (x^p_{q(k)}, x^q_{p(k)})\}$. Let

$$\mathcal{E} = \mathcal{E} \bigcup \{(x^p_{q(1)}, x^q_{p(1)}), \cdots, (x^p_{q(k)}, x^q_{p(k)})\} \tag{10}$$

25: **end for**
26: **end for**

manifolds [40], [41]. Traditional graph construction methods, k-NN and ε-NN, cannot guarantee the connectivity of the graph for multi-manifold data [42]. For example, when the data are spread among multiple clusters, a small neighborhood size for graph construction may leads to a disconnected graph, on which the geodesic distances between data points across the disconnected components cannot be estimated. Consequently, the data cannot be projected into a single low-dimensional coordinate system. In order to construct a better graph, we propose a new graph construction method, namely the k-Connectivity Graph (k-CG) method. The proposed method first builds a k-NN (or ε-NN) graph over the whole data in a semi-supervised manner, and then connects the adjacent graphs by the k shortest inter-manifold edges. Details of the method are presented in Algorithm 1 and explained bellow.

The k-CG method:

Step 1. *(Algorithm 1, lines 4–18) Construct the k-NN or ε-NN neighborhood graph in a semi-supervised manner.* Given an appropriate neighborhood size, define a graph G with the data points as the vertices by the means of k-NN or ε-NN method. For the training data with class labels, each data point is connected to its nearest neighbors in the same class; for the training data without class labels, each data point is connected to its nearest neighbors in the training set. It can be seen that the nearest neighbor approach cannot guarantee a connected graph. At this step, several disconnected graph components may be obtained and each graph component can be considered as a data manifold. It is assumed that there are P data manifolds and the p-th data manifold can be written as $\mathcal{S}^p = \{x_1^{(p)}, \ldots, x_{n_p}^{(p)}\}$.

Step 2. *(Algorithm 1, lines 19–21) Compute the average number of the neighbors.* If the ε-NN method is applied to define the graph G at Step 1, the average number k of the neighbors needs to be computed. Let l_i be the number of the neighbors of x_i. The value of k is set to be the nearest integer to $\sum_{i=1}^{N} l_i/N$.

Step 3. *(Algorithm 1, lines 22–26) Connect the k nearest inter-manifold data points among manifolds.* Identify the k nearest inter-manifold data pairs, $\{(x^p_{q(i)}, x^q_{p(i)}), i = 1, \ldots, k\}$, between $\mathcal{S}^p$ and $\mathcal{S}^q$, and connect these data pairs by edges, for $p, q = 1, \ldots, P$. Then the k-CG graph is constructed on $\mathcal{X}$.

The k-CG graph has the following three advantages on multi-manifold data:

1. It is connected by only short edges. Our k-CG method does not need to enlarge the neighborhood size to build a connected graph. Consequently, it can build a graph faithfully following the data manifold and avoid the "short-circuit" problem [43].
2. It has multiple edge connections among any partitions of the graph. Compared with the extended graph construction proposed in [38], our method uses k edge to connect different partitions. Therefore, the geodesic distances across the partitions can be better approximated.
3. The k-CG graph is constructed in a semi-supervised manner. Each labeled training data point is only connected to data points from the same class. Therefore, the graph distances among data points from the same class by our method are shorter than those by the k-NN or ε-NN method. This is an advantage for classification.

When k-CG graph is built, the lengths of the shortest paths among the data points can be computed by the classical Floyd-Warshall's or Dijkstra's algorithm. Let $d_G(x_i, x_j)$ be

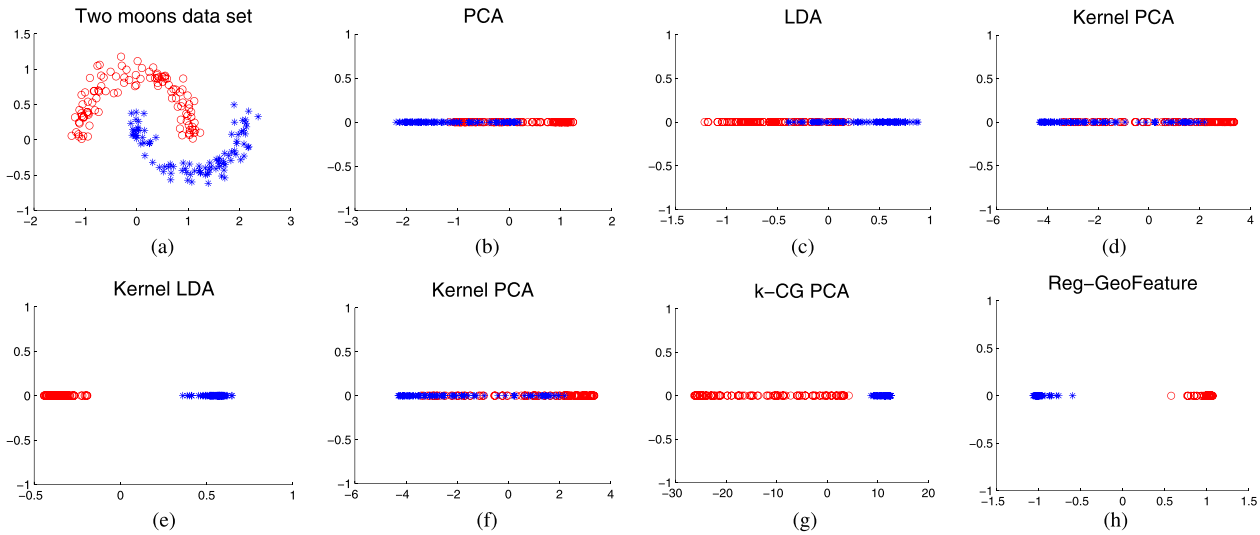


Fig. 2. 1-dimensional representation of the two moons dataset by (b) PCA (c) LDA (d) Kernel PCA (e) Kernel LDA (f) 12-NN feature PCA (g) 12-CG feature PCA and (h) Reg-GeoFeature.

the graph distance between data points x_i and x_j in the neighborhood graph and let the feature vector f_i be $f_i = (d_G(x_i, x_1), \ldots, d_G(x_i, x_N))^T$.

However, the batch mode graph construction method is computationally expensive for streaming data. We propose an incremental graph construction procedure for the new data points. When a new data point x comes, we first identify its k-nearest or ε-nearest neighbors in $\mathcal{X}$, which are assumed to be $\{x_1, \ldots, x_k\}$. Then, we set the edges between x and these neighboring points. In this way, the lengths of the shortest paths from x to the data points in $\mathcal{X}$ can be computed by

$$d_G(x, x_i) = \min_{t=1,\ldots,k} \{\|x - x_t\| + d_G(x_t, x_i)\}, \quad \text{for} \quad i = 1, \ldots, N. \tag{11}$$

Though this procedure may be less accurate than implementing Floyd-Warshall's or Dijkstra's algorithm on the new dataset $\mathcal{X} \cup \{x\}$, it has a low computational complexity. Only $O((k+1)N)$ computational time is needed to include each new data point. Then the feature vector of x is obtained as $f = (d_G(x, x_1), \ldots, d_G(x, x_N))^T$. In the appendix, we prove that the replacement is an injection and therefore, it is sensible for classification. In view of this property, we replace $\mathcal{X} = \{x_i, i = 1, \ldots, N\}$ with feature vectors $\mathcal{F} = \{f_i, i = 1, \ldots, N\}$.

To verify the claim that the feature vectors contain the discriminative nonlinear information, we generate a two-moon dataset, which is shown in Fig. 2(a). Two classes of labeled samples are shown with circles and stars and each class consists of 100 data points in $\mathbb{R}^2$. As can be seen, the dataset is linearly inseparable. So it is impossible to apply the linear dimension reduction algorithms to find its separable 1-dimensional representation of the data, as shown in Fig. 2(b) and (c). We test on a wide range of kernel width, but kernel PCA just cannot find the separable representation of the data, as can be seen in Fig. 2(d). On the other hand, Kernel LDA can obtain the separable 1-dimensional representation, as shown in Fig. 2(e). We build the feature vectors on the 12-NN graph, and reduce the dimension of feature vectors by PCA. As can be seen from Fig. 2(f), representations of the data points from different classes are still inseparable. Finally, we build feature vectors on the 12-CG graph and reduce the dimension of feature vectors by PCA and our Reg-S3DR(will be presented later). The results given in Fig. 2(g) and (h) indicate that the low-dimensional representations are linearly separable.

B. A Regularized Sparsity Preserving Approach for Semi-Supervised Dimension Reduction

In this subsection, we propose a novel Regularized Sparsity preserving Semi-Supervised Dimension Reduction (Reg-S3DR) method for feature vectors. The proposed method preserves the relationship between data samples, which emerge as a sparsely defined weighted neighborhood graph, in the manner of a regularized regression problem. As the third step of our Reg-GeoFeature algorithm, we apply it to feature vectors $\mathcal{F}$ instead of the original dataset $\mathcal{X}$.

To realize robust dimension reduction, we propose to utilize the following regularized regression model,

$$\Phi^* = \arg\min_{\Phi} \left\{ \frac{1}{l} \sum_{i=1}^{l} V(f_i, y_i^{label}, \Phi) + \gamma_K \|\Phi\|_K^2 + \gamma_I \|\Phi\|_I^2 \right\}, \tag{12}$$

where the optimization variable is a vector mapping $\Phi : \mathbb{R}^N \to \mathbb{R}^d$, $y_i^{label} \in \mathbb{R}^d, i = 1, \ldots, l$, denote the prior low-dimensional representations which are unknown and to be constructed from the class labels of the data points, and γ_K and γ_I are two regularization parameters. Here, $V(f_i, y_i^{label}, \Phi) =$

$\|\Phi(f_i) - y_i^{label}\|_2^2, i = 1, \ldots, l$, is the squared loss function, $\|\Phi\|_K^2$ is a measurement of the complexity of mapping Φ and $\|\Phi\|_I^2$ measures the ability of Φ in preserving certain structure of data. In this paper, the structure of data is represented as a sparsely defined weighted neighborhood graph.

How to construct $\{y_i^{label}\}_{i=1}^l$ of the labeled data samples is a key issue of our proposed Reg-S3DR method. The low-dimensional representations of data from different classes should spread as far as possible while low-dimensional representations from the same class should be located as close as possible. To do this, we apply the method proposed in [44], which generates random label vectors for data samples from different classes. Firstly, C vectors $\{L_k\}_{k=1}^C \subset \mathbb{R}^d$ are randomly generated. Secondly, we set the label vectors to labeled data points

$$y_i^{label} = L_k, (i = 1 \ldots, l), \quad \text{if } x_i \text{ is in the } k\text{-th class.}$$

Each component of L_k is a random number from the $[0, 1]$ uniform distribution. It has been proven that the probability of the label vectors spread far apart from each other is very high. The following theorem can be utilized to corroborate this claim.

Theorem 3.1: Since $\{L_k\}_{k=1}^C \subset \mathbb{R}^d$ are uniformly distributed in a d-dimensional unit hypercube, the probability that all the other $C - 1$ vectors are not in $B_k(r)$ is $(1 - r^d)^{C-1}$, where $0 < r < 1$ and $B_k(r)$ represents a d-dimensional hypersphere of radius r centered around L_k.
For example, let $r = 0.5$, $d = 10$ and $C = 10$, the probability that no other $C - 1$ label vectors in $B_k(r)$ is 99.12%.

According to the theory of statistical learning [49], [50], the regularizer $\|\Phi\|_K^2$ is usually defined as the norm of function in certain Reproducing Kernel Hilbert Space (RKHS). Given a positive semi-definite kernel $k(u, v)$, there is an associated RKHS $\mathcal{H}_K$. Any function $\phi_s \in \mathcal{H}_K$ can be expressed as a linear combination of kernel functions $\phi_s(\cdot) = \sum_i \eta_i^s k(u_i, \cdot)$. For a vector mapping $\Phi = [\phi_1, \ldots, \phi_d]^T$, $\|\Phi\|_K^2$ can be defined as:

$$\|\Phi\|_K^2 = \sum_{s=1}^d \|\phi_s\|_K^2 = \sum_{s=1}^d \sum_{i,j} \eta_i^s \eta_j^s k(u_i, u_j). \tag{13}$$

To make the mapping Φ more discriminative, the regularizer $\|\Phi\|_I^2$ should be defined using both the geometrical structure of the data manifold and the label information. In doing so, the pairwise weights between data points are defined in the following:

$$W_{ij} = \begin{cases} \kappa, & \text{if } x_i \text{ and } x_j \text{ belong to the same class,} \\ 1, & \text{if } x_i \text{ or } x_j \text{ is unlabeled but they are} \\ & \text{neighbors on our } k\text{-CG graph,} \\ -\kappa, & \text{if } x_i \text{ and } x_j \text{ belong to different classes,} \\ & \text{and they are neighbors on } k\text{-CG graph,} \\ 0, & \text{otherwise.} \end{cases}$$

The value of W_{ij} represents the prior knowledge of whether x_i and x_j are from the same class. A large W_{ij} means that the confidence that x_i and x_j are from the same class is high, while a negative W_{ij} means that the confidence that x_i and x_j are from different classes is high. $W_{ij} = 0$ means that we have no prior knowledge on the label relation between x_i and x_j. Therefore, the value of κ is supposed to be relatively large. In our experiments, κ is empirically set to be 5. A reasonable criterion for finding a discriminative mapping is to optimize the objective function

$$\min \sum_{i=1}^N \sum_{j=1}^N W_{ij} \|\Phi(f_i) - \Phi(f_j)\|^2. \tag{14}$$

This objective function incurs a heavy penalty if x_i and x_j are mapped far apart when W_{ij} is large. We define the manifold smoothing regularizer $\|\Phi\|_I^2$ as

$$\|\Phi\|_I^2 = \frac{1}{2N^2} \sum_{i=1}^N \sum_{j=1}^N W_{ij} \|\Phi(f_i) - \Phi(f_j)\|^2. \tag{15}$$

As can be seen, we can maximize the distances between low-dimensional representations from different classes and minimize the distances between representations from the same class by minimizing $\|\Phi\|_I^2$.

Then, our Reg-S3DR method is presented as follows:

$$\Phi^* = \underset{\phi_s \in \mathcal{H}_K, s=1\ldots,d}{\arg\min} \left\{ \frac{1}{l} \sum_{i=1}^l \|\Phi(f_i) - y_i^{label}\|_2^2 + \gamma_K \|\Phi\|_K^2 + \frac{\gamma_I}{2N^2} \sum_{i=1}^N \sum_{j=1}^N W_{ij} \|\Phi(f_i) - \Phi(f_j)\|^2 \right\}. \tag{16}$$

Based on the following theorem, we can derive the explicit function solution of (16).

Theorem 3.2: [44] The minimizer of optimization problem (16) admits an expansion

$$\Phi^*(f) = \sum_{i=1}^N \alpha_i k(f_i, f), \tag{17}$$

in terms of the labeled and unlabeled data points, where $\alpha_i = [\alpha_{1i}, \ldots, \alpha_{di}]^T \in \mathbb{R}^d$, f is a feature vector built by geodesic distances, and $k(\cdot, \cdot)$ is some kernel function.

Substituting the terms given in Eqs. (13), (15), and the expansion (17) into (16), and by matrix manipulations, we can formulate (16) in the form:

$$A^* = \underset{A \in \mathbb{R}^{d \times N}}{\arg\min} \left\{ \frac{1}{l} \mathrm{tr}\left((AKJ - Y^{label})(AKJ - Y^{label})^T \right) + \gamma_K \mathrm{tr}\left(AKA^T \right) + \frac{\gamma_I}{N^2} \mathrm{tr}\left(AKLKA^T \right) \right\}, \tag{18}$$

where $A = [\alpha_1, \ldots, \alpha_N] \in \mathbb{R}^{d \times N}$ is the coefficient matrix for the expansion of Φ, $Y^{label} = [y_1^{label}, \ldots, y_l^{label}, \mathbf{0}, \ldots, \mathbf{0}] \in \mathbb{R}^{d \times N}$ is the target matrix, $K = \left(k(f_i, f_j)\right) \in \mathbb{R}^{N \times N}$ is the kernel matrix, $L = S - W \in \mathbb{R}^{N \times N}$ denotes the graph Laplacian matrix, $W = \left(W_{ij}\right) \in \mathbb{R}^{N \times N}$, S is a diagonal matrix whose diagonal element $S_{ii} = \sum_{j=1}^N W_{ij}$, I is the identity matrix, and $J \in \mathbb{R}^{N \times N}$ is a diagonal selection matrix whose first l diagonal elements are ones and the rest diagonal elements are zeros. Based on their definitions, we have $J^T = J$, $J^2 = J$, $Y^{label} = Y^{label} J$ and kernel matrix K

is a positive definite matrix. Imposing the derivative of the objective function in (18) with respect to A to zero, it follows that

$$\frac{1}{l}\left(Y^{label} - AKJ\right)(KJ)^T + \gamma_K AK + \frac{\gamma_I}{N^2} AKLK = 0. \quad (19)$$

By matrix manipulations, Eq. (19) can be formulated as

$$A(KJ - \gamma_K lI - \frac{l}{N^2}\gamma_I KL) = Y^{label}.$$

Subsequently, the least squares solution of (19) can be obtained as

$$A^* = Y^{label}\left(KJ + \gamma_K lI + \frac{\gamma_I l}{N^2} KL\right)^{-1}. \quad (20)$$

Then the desired dimension reduction mapping is given as

$$Z = \Phi^*(f) = \sum_{i=1}^{N} \alpha_i^* k(f_i, f),$$

where α_i^* is the i-th column of A^*. As geodesic distance is regarded as a kernel, we simply use linear kernel for dimension reduction mapping Φ, i.e., $k(u_i, u_j) = u_i^T u_j$.

C. The Regularized Geodesic Feature Learning Algorithms

By including the k-CG graph construction method, the replacement of the original data with the feature vectors, and the Reg-S3DR method, our Reg-GeoFeature algorithm has three steps.

Algorithm 3.2. (The Reg-GeoFeature Algorithm)

Step 1. *Construct a connected graph.* Construct a neighborhood graph over $\mathcal{X}$ using the k-CG method in Algorithm 1. A weighted graph $G = \{\mathcal{X}, D\}$ is constructed, where $(D)_{ij} = \|x_i - x_j\|$ if x_i and x_j are connected by an edge and $(D)_{ij} = \infty$ otherwise.

Step 2. *Compute feature vectors.* Compute the lengths of pair-wise shortest paths on the graph by implementing the Floyd-Warshall's or Dijkstra's algorithm, and then replace x_i with the feature vector $f_i = [d_G(x_i, x_1), \ldots, d_G(x_i, x_N)]^T$, for $i = 1, \ldots, N$. The class label of f_i is set to be y_i, for $i = 1, \ldots, l$.

Step 3. *Compute d-dimensional embedding.* Apply the Reg-S3DR method on the feature vectors. Let the computed coefficients for the expansion be A. Each data point x_i is represented by its low-dimensional vector $z_i = AF^T f_i$, where $F = [f_1, \ldots, f_N]$.

Algorithm 3.2 presents the Reg-GeoFeature algorithm, which trains an explicit dimension reduction mapping for feature vectors of both the labeled and unlabeled data. When the low-dimensional representations of data points are obtained, one can train an efficient classifier using the labeled low-dimensional representations.

In the following, we propose the online Reg-GeoFeature algorithm for test data. Given an unlabeled sample x, online Reg-GeoFeature first maps it to low-dimensional space, and then applies the trained classifier to its low-dimensional representation.

Algorithm 3.3. (Online Reg-GeoFeature Algorithm)

Step 1. Compute pair-wise Euclidean distances $\|x - x_i\|$, for $i = 1, \ldots, N$. Identify the k-nearest neighbors or ε-nearest-neighbors of x, which are assumed as $\{x_1, \ldots, x_k\}$.

Step 2. Compute the lengths of the shortest paths for x by Eq. (11). Then, the feature vector of x is given as $f = [d_G(x, x_1), \ldots, d_G(x, x_N)]^T$.

Step 3. Then, we obtain the low-dimensional representation of x as $z = AF^T f$, where $F = [f_1, \ldots, f_N]$.

D. Time Complexity Analysis

The time complexity of the Reg-GeoFeature algorithm is a crucial issue in its applications. The graph construction process, which applies the k-CG method, needs $O(kN^2)$ computational time. Subsequently, Dijkstra's algorithm consumes $O(kN \log N)$ computational time on the graph. The complexity for solving the matrix inversion problem (20) directly is $O(N^3)$. Therefore, the total time complexity of the Reg-GeoFeature algorithm is $O(kN^2 + kN \log N + N^3)$.

For online classification tasks, there are three steps in the algorithm. The first step which computes pair-wise Euclidean distances consumes $O(N)$ computational time. Lengths of the shortest paths are computed at the second step, which needs $O(kN)$ computational time. The time complexity for the final projection is $O(N)$. Therefore, for online classification, the Online Reg-GeoFeature algorithm consumes $O((k+2)N)$ computational time for a new point.

E. Differences From the Previous Work

Cai et al. [25] proposed the semi-supervised discriminant analysis based on the LDA algorithm. Independently, Song et al. [26] proposed similar semi-supervised dimension reduction algorithms based on LDA and MMC. Our algorithm is intrinsically different from their algorithms in two aspects. First, the dimension reduction mappings of their algorithms [25], [26] are obtained by solving eigen-decomposition problems, whereas our algorithm takes the regression manner. Second, the nonlinear algorithms [25], [26] by Cai et al. and Song et al. are achieved through the kernel trick, while our algorithm extracts the multi-manifold structure of the data in the form of feature vectors. In the experiments of this paper, we compare the proposed Reg-GeoFeature algorithm with the Kernel Semi-supervised LDA, which is named as SS-KDA in [25].

Our work is also different from the E-Isomap algorithm [38]. E-Isomap applies k-NN method to build a graph, which ignores class information. Besides, it achieves dimension reduction using the classical LDA method, which suffers from the problems such as singularity of within-class scatter matrices and the limited available projection directions.

In this paper, Reg-S3DR replaces the Semi-Supervised Discriminant Analysis (SSDA) method of [17]. The merits of Reg-S3DR can be summarized as follows:

1. *The labels of data are sufficiently utilized by the regression model.* The label information is not only used to

TABLE II
BRIEF DESCRIPTIONS OF THE COMPARING ALGORITHMS

Property	Algorithms
Linear	PCA, LDA, SDONPP, IsoProjection, LSDA, MMP
Nonlinear	SS-KDA, E-Isomap, Multi-MDA, Reg-GeoFeature
Supervised	LDA, LSDA, IsoProjection, E-Isomap
Semi-Supervised	SS-KDA, SDONPP, MMP, Multi-MDA, Reg-GeoFeature

TABLE III
RELEVANT ATTRIBUTES OF THE FOUR DATASETS

Dataset	Size	# of features	# of classes
MNIST [47]	4157	784	4
USPS [46]	4400	256	4
ITS [45]	2978	576	2
MIT CBCL [48]	2000	361	2

generate the prior low-dimensional representations of labeled data, but also used to define the graph weights for the manifold smoothing regularizer. This makes our regression model very discriminative.

2. *The regression model is more robust than SSDA*. This is because the complexity of dimension reduction mapping, the data structure, and the discriminative information can be naturally incorporated into the regression model as regularizers. Both ill-posedness and over-fitting issues can thus be mitigated.

IV. EXPERIMENTS

In this section, we make use of four publicly available datasets, namely, MNIST [47], USPS [46], Intelligent Traffic System (ITS) [45], and MIT CBCL [48]. Among them, MNIST and USPS are handwritten digital image datasets. ITS contains images with human and images without human, which is obtained from intelligent traffic systems. MIT CBCL contains human face images and non-face images. The important statistics of these datasets are summarized in Table III. We compare the proposed Reg-GeoFeature algorithm with representative dimension reduction algorithms, PCA [4], LDA [2], SS-KDA [25], SDONPP [13], IsoProjection [18], LSDA [20], MMP [19], E-Isomap [38] and our previous work Multi-MDA [17]. The properties of the comparing algorithms are summarized in Table II.

A. Dataset Description

MNIST[1] [47] is a benchmark dataset for digital image classification. Each sample is a 28×28 image of a handwritten digit which can be transformed to a 784-dimensional data point. The subset used in this paper consists of 980 samples from class '0', 1135 samples from class '1', 1032 samples from class '2', and 1010 samples from class '3', which form a dataset with four classes. Samples of this dataset are shown in Fig. 3.

USPS[2] [46] is another benchmark handwritten digit dataset, which contains 1100 samples for each class from '0', '1' to '9'. Each sample of this dataset is a 16×16 image of a handwritten digit and can be transformed to a 256-dimensional data point. The dataset used in our experiment consists of the samples from classes '0', '1', '2' and class '3', which form a four class dataset with 4400 data samples. Samples of this dataset are shown in Fig. 4.

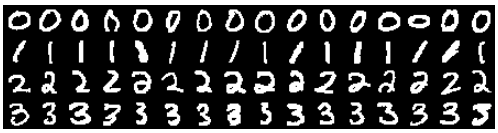


Fig. 3. Image samples from the MNIST handwritten dataset of classes '0', '1', '2' and '3'.

Fig. 4. Image samples from the USPS handwritten dataset of classes '0', '1', '2' and '3'.

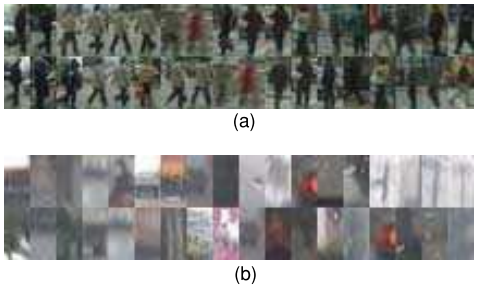


Fig. 5. (a) Images samples with human, (b) Image samples without human.

Fig. 6. Image samples from the CBCL dataset. The first two rows show some face images, and the last two rows show some non-face images.

The ITS [45] dataset is collected by a camera on a moving car. Each sample is a cropped $24 \times 12 \times 3$ color image with or without a person in it and can be transformed to a 576-dimensional data point. There are 950 samples of people walking or running, and there are 2028 samples without a person. Samples with people in them are shown in Fig. 5(a) and samples without people are shown in Fig. 5(b).

MIT CBCL[3] [48] dataset contains 2429 face images and 4548 non-face images. Each image has 19×19 pixels and is transformed to a 361-dimensional vector. This dataset contains two classes of data points, face and non-face. In the experiment, we use a subset of this dataset, which comprises 1500 face and 1500 non-face images. In Fig. 6, we show some face and non-face images contained in this dataset.

[1] http://yann.lecun.com/exdb/mnist/

[2] http://www.cs.nyu.edu/~roweis/data.html

[3] http://cbcl.mit.edu/software-datasets/FaceData2.html

B. Experimental Settings

We apply 10-fold cross-validation to evaluate the comparing algorithms. Given any dataset, we randomly split it into ten equally-sized subsets. For the k-th subset, the other 9 subsets form the data $\mathcal{X}$ and the k-th subset is used as the test data $\mathcal{X}_{test}$. In each class of $\mathcal{X}$, we label α percent of the data points and denote all labeled data in $\mathcal{X}$ as $\mathcal{X}_{label}$. Therefore, $\mathcal{X}$ can be used as a partially labeled training set and $\mathcal{X}_{label}$ is a totally labeled training set. For unsupervised and semi-supervised methods, PCA, SS-KDA, SDONPP, MMP, Multi-MDA, and Reg-GeoFeature, $\mathcal{X}$ is applied to learn the projection mappings. For supervised methods, LDA, LSDA, E-Isomap, and IsoProjection, $\mathcal{X}_{label}$ is applied to train the projection mappings.

Classification on the unlabeled samples in $\mathcal{X}$ is conducted as follows:

Step 1. Train an explicit dimension reduction mapping using the training data. We apply the algorithms to $\mathcal{X}$ or $\mathcal{X}_{label}$, which provide explicit dimension reduction mappings for unlabeled data in $\mathcal{X}$ and the test data points. Assume that $\mathcal{Z} = \{(z_i, y_i), z_{l+j}, i = 1, \ldots, l, j = 1, \ldots, u\}$ is the low-dimensional representation of $\mathcal{X} = \{(x_1, y_1), \ldots, (x_l, y_l), x_{l+1}, \ldots, x_{u+l}\}$.

Step 2. Considering $\{(z_i, y_i), i = 1, \ldots, l\}$ as the training set, we implement the nearest neighbor classifier on the unlabeled set $\{z_{l+j}, , j = 1, \ldots, u\}$.

Classification on the test dataset $\mathcal{X}_{test}$ is conducted as follows:

Step 1. Apply the trained dimension reduction mappings on $\mathcal{X}_{test}$, where the computed low-dimensional representations are assumed as $\{z_j^{test}, j = 1, \ldots, T\}$.

Step 2. Considering $\{(z_i, y_i), i = 1, \ldots, l\}$ as the training set, we implement the nearest neighbor classifier on the test set $\{z_j^{test}, j = 1, \ldots, T\}$.

The codes for IsoProjection, LSDA, MMP, LDA and SS-KDA methods are downloaded from the web.[4] We implemented the other methods ourselves and tuned the parameters for each method for a fair comparison. For SS-KDA, SDONPP, Multi-MDA and Reg-GeoFeature, the regularization parameters need to be set beforehand to balance different terms. For fair comparisons, we set each parameter to $\{10^{-8}, 10^{-5}, 10^{-3}, 10^{-2}, 1, 10, 10^2, 10^3\}$, and then choose the parameter configuration corresponding to the top-1 recognition accuracy. The Gaussian Kernel $\exp\{-\|x-y\|^2/\sigma^2\}$ is used for SS-KDA, and σ is set as $\sigma = 2^{(n-10)/2.5}\sigma_0$, $n = 0, \ldots, 20$, where σ_0 is the standard derivation of pairwise Euclidean distances between training samples. The top-1 recognition accuracy of the best parameter configuration is reported. By referring to [13], the number k' of the selected neighbors having the same label for SDONPP algorithm is set as 5. The neighborhood size k for SDONPP, SS-KDA, E-Isomap, MMP, IsoProjection, and LSDA is chosen between 4 and $8C$ at a sampling intervals of 2. The neighborhood size k for multi-MDA and Reg-GeoFeature is set as 10, which is an empirically good choice on the four datasets.

[4]http://www.cad.zju.edu.cn/home/dengcai/Data/data.html

For LDA, E-Isomap and SS-KDA algorithms, there are at most $C-1$ nonzero generalized eigenvectors [25], [38]. So, an upper bound on dimension d is $C-1$, where C is the number of classes. That explains why the experimental results of LDA, SS-KDA, and E-Isomap are unaffected by varying dimensions.

In the following, the averaged cross-validation recognition rates with standard variations are reported accordingly.

C. Experiments on the Techniques Applied in Reg-GeoFeature Algorithm

It is necessary for us to verify the effectiveness of our proposed techniques: geodesic distance feature vectors vs. original data points, i.e., Reg-GeoFeature vs. Reg-S3DR; our new term $\|\Phi\|_I^2$ vs. the Laplacian smoothing term [50], which is unsupervised and built by local similarity, i.e., Reg-S3DR vs. Laplacian Regularized Least Squares Classifier (LapRLSC) [50] with linear kernel.

In order to make a fair comparison, the target dimension d is set as 50 for all datasets, the neighborhood size k is set as 10 for all four methods and the regularization parameters are set to their best performances. And the percentage of the labeled training data, α, varies from 5 to 40. In Fig. 7, the averaged recognition accuracies and standard variations on both unlabeled training data and test data are reported.

According to the results shown in Fig. 7, we have the following observations.

1. Reg-GeoFeature shows higher classification accuracies than Reg-S3DR. This indicates geodesic distance could be regarded as a kind of kernel, which unfolds the nonlinear structure and thus makes originally inseparable data linearly separable in the distance space.
2. Reg-S3DR has better performances than linear LapRLSC. This implies that the regularizer $\|\Phi\|_I^2$ contains more discriminative information than the Laplacian based manifold smoothing regularizer [50].

D. Classification With Varying Dimensions

Fixing $\alpha = 10$, which means 10 percent of the data in $\mathcal{X}$ are labeled, we evaluate the performances of comparing algorithms with different dimensions. Here the dimension varies from C to $70 + C$, where C is the number of classes. The results of the comparing methods on unlabeled data are reported in Fig. 8, and the results of the methods on test data are reported in Fig. 9.

1) MNIST: Figs. 8(a) and 9(a) give the curves of the classification accuracies on unlabeled data and test data of MNIST dataset under different reduced dimensions. Reg-GeoFeature shows the best classification performance. Except Reg-GeoFeature, Multi-MDA has slightly better performance than the other methods when $d \geq 10$. The classification accuracy curve of E-Isomap is the lowest as there are limited number of labeled data points to approximate pairwise geodesic distances.

2) USPS: The results of comparing methods on unlabeled training data and test data of USPS dataset are reported in Figs. 8(b) and 9(b). As can be seen, Reg-GeoFeature outperforms other methods on all the dimensions. Multi-MDA

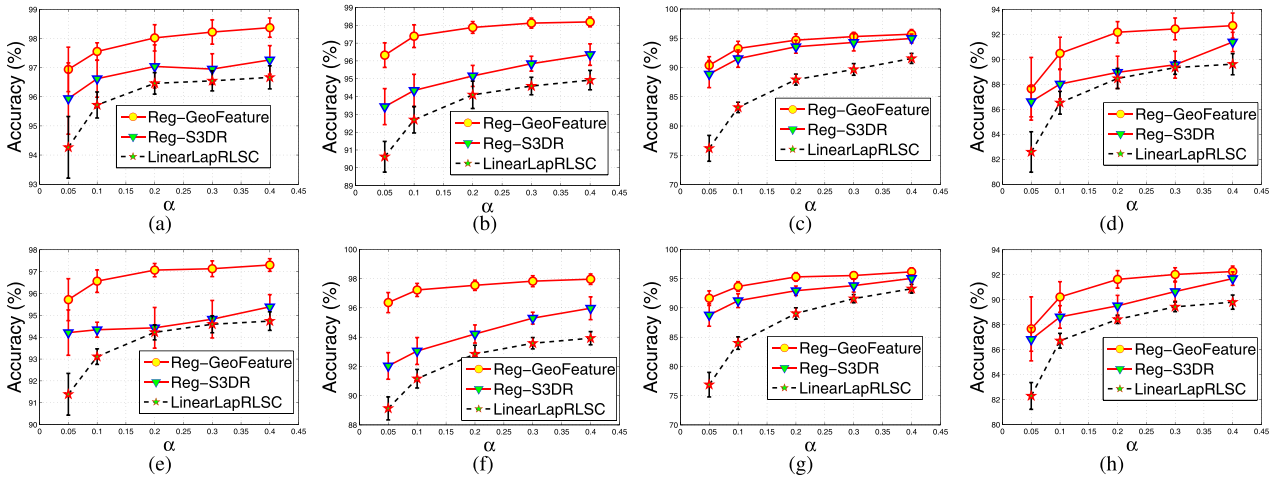


Fig. 7. Classification results of Reg-GeoFeature, Reg-S3DR, and LapRLSC with linear kernel methods on (a) the unlabeled data of MNIST, (b) the unlabeled data of USPS, (c) the unlabeled data of ITS, (d) the unlabeled data of CBCL, (e) the test data of MNIST, (f) the test data of USPS, (g) the test data of ITS, (h) the test data of CBCL.

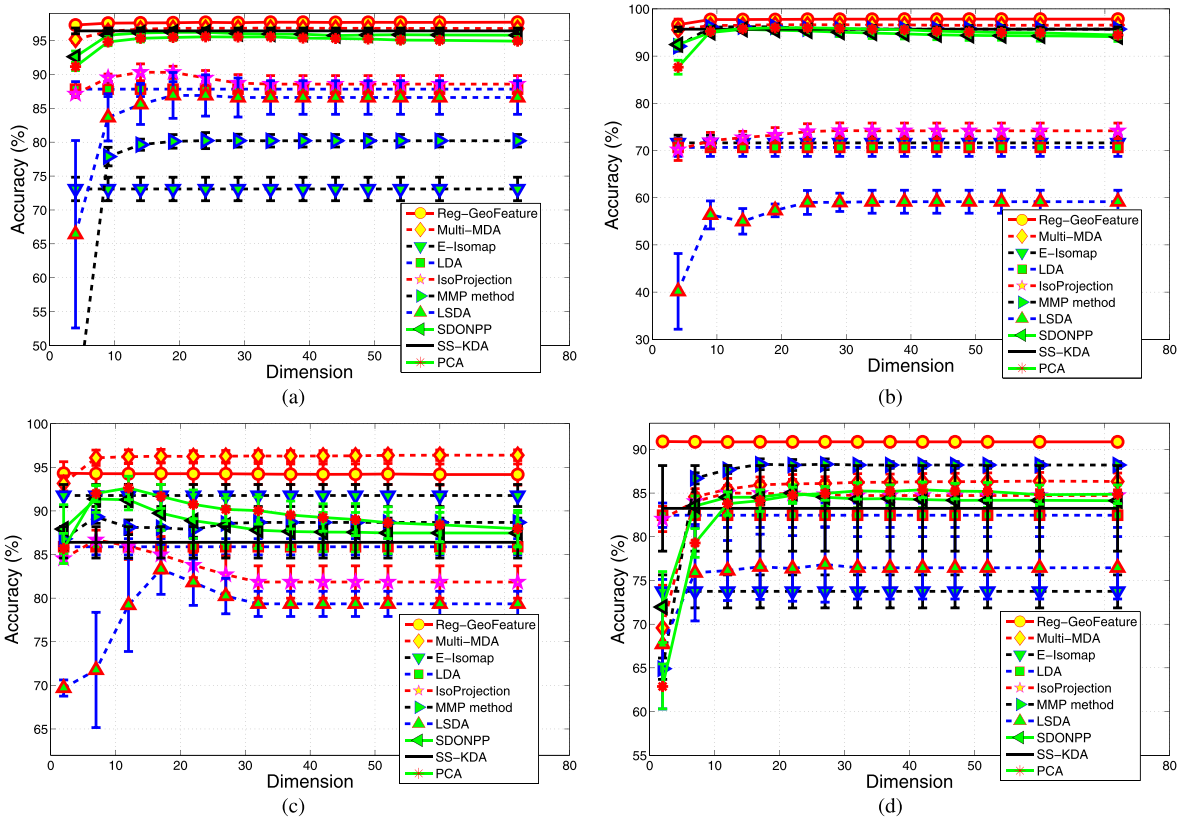


Fig. 8. Classification results of the comparing algorithms on unlabeled data in $\mathcal{X}$ with various dimension d, where d changes from C to 70+C. (a) MNIST, (b) USPS, (c) ITS, and (d) MIT CBCL.

has the second highest classification accuracies on this dataset. SS-KDA ranks the third on unlabeled training data but shows poorer results on the test data. The reason for the poorer results of SS-KDA is that the kernel width works well on training data is unsuitable for test data, which is unseen in the training stage.

3) ITS: As can be seen from Figs. 8(c) and 9(c), Multi-MDA achieves a gain in accuracy of approximately 3 percent on ITS dataset. Reg-GeoFeature shows the second highest recognition rates. LDA and SS-KDA have poor performance on this dataset for two reasons: firstly, there are limited projection directions for the two class data. Secondly, the numbers of data points in different classes are uneven, i.e., 950 vs. 2028. So the learned dimension reduction mapping may have a bias toward one of the classes.

4) MIT CBCL: Figs. 8(d) and 9(d) present the comparisons of the recognition rates on unlabeled data and test data of CBCL dataset under different reduced dimensions. As can be

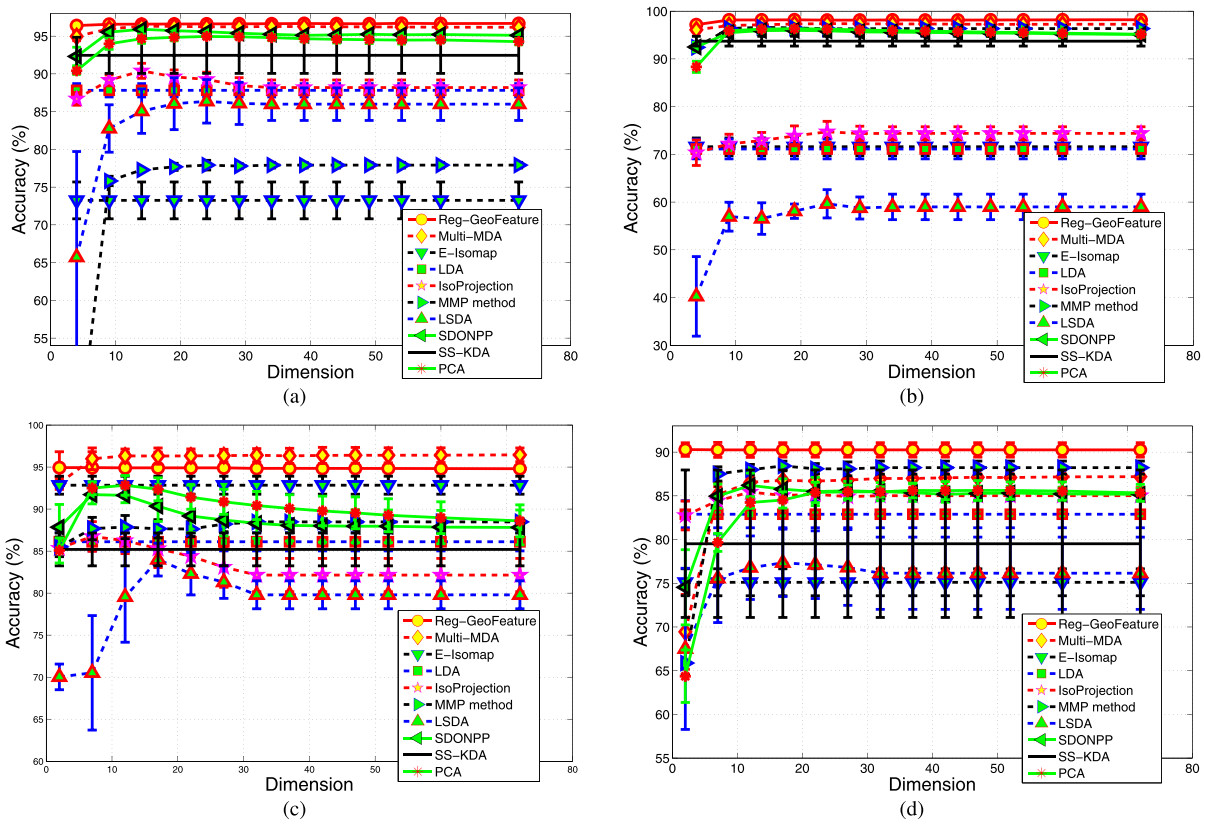


Fig. 9. Classification results of the comparing algorithms on test data $\mathcal{X}_{test}$ with various dimension d, where d changes from C to $70 + C$. (a) MNIST, (b) USPS, (c) ITS, and (d) MIT CBCL.

seen, Reg-GeoFeature outperforms other methods and MMP has better performance than Multi-MDA. Still, the recognition accuracies of Reg-GeoFeature are always 3 or 4 percent higher than the MMP method. E-Isomap and LSDA show the poorest performances on this dataset.

As can be seen from the experimental results in Figs. 8 and 9, the performance of Reg-GeoFeature seems unaffected by the variation of reduced dimensions. This is interesting and similar phenomenon can be found in the paper [44]. For dimension reduction methods in regression manner, it seems that the absolute distances among the targets of different classes do not matter. But they should spread evenly, i.e., the distances among them should be nearly equal, which is satisfied by our label/target generation method.

E. Classification With Different Portions of Labeled Training Samples

We conduct experiments on datasets with fixed target dimension d, where $d = 50$ for the most of the comparing methods and a suitable value (C or $C - 1$) for LDA, SS-KDA, and E-Isomap. Meanwhile, the percentage α of labeled training data changes from 5, 20, 30 to 40. When α increases, the number of labeled data $\mathcal{X}_{label} \subset \mathcal{X}$ increases accordingly.

The classification results on MNIST and USPS datasets are reported in Table IV, while the results on ITS and CBCL datasets are reported in Table V. As can be seen, when $\alpha \leq 20$, Reg-GeoFeature and Multi-MDA show the best performance on all of these datasets, which implies that geodesic distance feature is discriminative for semi-supervised classification. However, the gains of our method over other methods decrease as the number of labeled training samples increases. This is because, in order to get better results, the regularization parameters for Reg-GeoFeature should be updated to the varying number of labeled training data points. We simply tune these parameters for each method on a dataset when $\alpha = 10$. So the parameters are invariant to the number of training data points. Multi-MDA is generally better than other methods on the ITS dataset, except for the case when $\alpha = 30$ and 40 and on the test data. On MIT CBCL dataset, the results of Reg-GeoFeature and Multi-MDA are not the best in all comparisons. But the geodesic feature based methods still show competitive performances on CBCL dataset. Especially when the number of labeled training data points is relatively small.

F. Discussion

According to the experiments systematically performed on the datasets, we have several observations:

1. PCA and SDONPP show better performance than LDA, E-Isomap, and SS-KDA on the applied datasets. There are two reasons for this. First, the available projection directions of LDA, E-Isomap, and SS-KDA are limited by the number of classes, which are 4, 4, 2, 2

TABLE IV

CLASSIFICATION WITH DIFFERENT PERCENTAGES OF LABELED TRAINING DATA ON MNIST AND USPS DATASETS

	MNIST				USPS			
Unlabeled	α=5	α=20	α=30	α= 40	α=5	α=20	α=30	α=40
Reg-GeoFeature	**96.70(0.66)**	**97.98(0.37)**	**98.02(0.35)**	**98.09(0.40)**	**97.32(1.08)**	**98.87(0.21)**	**99.27(0.41)**	**99.54(0.17)**
Multi-MDA	95.54(1.06)	97.33(0.31)	97.63(0.46)	97.72(0.30)	95.57(1.32)	97.54(0.34)	97.95(0.32)	97.83(0.35)
E-Isomap	62.20(2.57)	77.15(0.93)	81.31(2.12)	83.09(2.73)	60.39(3.13)	77.45(1.79)	83.11(1.56)	83.70(1.99)
LDA	88.70(0.98)	67.14(2.66)	70.44(1.25)	48.03(15.1)	84.88(1.06)	87.81(1.37)	91.78(0.97)	93.13(1.28)
IsoProjection	85.07(1.24)	71.76(2.57)	72.82(2.88)	54.05(15.0)	83.18(2.50)	89.01(0.99)	93.24(1.01)	94.98(0.62)
MMP	70.91(1.66)	86.90(0.77)	90.21(0.86)	92.26(0.76)	94.74(0.65)	97.25(0.45)	97.62(0.29)	97.67(0.44)
LSDA	85.55(1.68)	56.60(4.77)	62.14(5.13)	52.78(16.4)	81.98(1.68)	90.44(1.16)	95.30(0.60)	96.43(0.53)
SDONPP	93.79(0.50)	96.95(0.28)	97.55(0.34)	97.85(0.35)	93.09(0.58)	96.44(0.40)	97.22(0.37)	97.68(0.33)
SS-KDA	95.96(0.61)	96.73(0.45)	97.72(0.35)	97.00(0.28)	94.42(2.28)	96.50(0.20)	96.75(0.24)	96.86(0.25)
PCA	92.78(0.37)	96.10(0.44)	96.68(0.40)	96.85(0.35)	93.42(0.95)	96.72(0.14)	97.43(0.38)	97.75(0.38)
Test	α=5	α=20	α=30	α= 40	α=5	α=20	α=30	α=40
Reg-GeoFeature	**95.89(0.10)**	**97.17(0.47)**	**97.38(0.48)**	97.50(0.32)	**97.48(0.71)**	**98.67(0.21)**	**98.72(0.30)**	**98.92(0.31)**
Multi-MDA	95.27(0.84)	96.78(0.18)	97.06(0.37)	97.25(0.38)	96.23(1.05)	97.85(0.56)	98.16(0.49)	98.31(0.51)
E-Isomap	62.47(2.29)	76.27(0.83)	80.45(1.16)	81.85(1.87)	60.78(2.73)	77.50(2.10)	83.20(3.22)	84.30(1.04)
LDA	88.23(0.95)	66.18(3.74)	69.82(2.77)	48.54(13.8)	84.26(1.34)	87.35(1.00)	91.82(0.94)	93.15(0.55)
IsoProjection	84.42(0.91)	71.25(2.22)	72.67(2.25)	54.92(14.3)	83.62(2.56)	88.92(0.80)	93.31(0.74)	95.00(0.80)
MMP	68.33(1.55)	83.88(0.84)	86.56(0.65)	88.20(0.30)	94.82(0.93)	97.40(0.55)	97.79(0.40)	98.10(0.40)
LSDA	85.34(2.38)	56.37(5.07)	61.89(4.70)	52.73(15.2)	81.72(2.10)	90.54(1.65)	95.20(0.72)	96.90(0.54)
SDONPP	93.21(0.51)	96.28(0.42)	97.05(0.31)	**97.71(0.30)**	93.60(1.00)	97.12(0.49)	97.75(0.21)	98.12(0.08)
SS-KDA	91.28(2.75)	91.15(3.59)	90.53(2.16)	92.10(1.60)	91.94(5.88)	96.32(0.62)	96.16(0.42)	96.63(0.67)
PCA	92.45(0.72)	95.82(0.43)	96.38(0.40)	96.87(0.44)	93.63(0.82)	97.29(0.49)	97.97(0.21)	98.32(0.17)

TABLE V

CLASSIFICATION WITH DIFFERENT PERCENTAGES OF LABELED TRAINING DATA ON ITS AND CBCL DATASETS

	ITS				CBCL			
Unlabeled	α=5	α=20	α=30	α= 40	α=5	α=20	α=30	α=40
Reg-GeoFeature	93.05(2.52)	96.26(0.48)	96.86(0.59)	97.70(0.49)	**88.94(1.13)**	**92.18(0.77)**	**92.85(0.62)**	**93.22(0.33)**
Multi-MDA	**94.43(2.24)**	**97.44(0.55)**	**97.52(0.27)**	**98.10(0.29)**	81.62(2.31)	89.80(0.25)	91.21(0.96)	91.91(1.43)
E-Isomap	84.16(4.27)	94.53(1.27)	95.78(0.88)	96.15(1.31)	66.98(4.56)	80.20(2.33)	83.02(0.51)	84.22(0.81)
LDA	86.76(2.51)	85.68(1.32)	84.96(1.27)	83.42(1.95)	82.27(2.72)	72.66(1.25)	74.93(0.40)	83.11(1.23)
IsoProjection	77.55(4.05)	86.56(0.97)	88.55(0.90)	87.74(1.45)	82.87(2.67)	73.96(1.55)	77.65(1.36)	85.91(1.16)
MMP	86.18(1.73)	89.53(0.90)	90.02(0.77)	91.58(1.46)	83.56(1.51)	91.66(0.64)	92.38(0.61)	92.55(0.80)
LSDA	74.37(1.34)	83.47(2.26)	82.64(2.48)	76.33(5.28)	79.36(2.22)	61.30(2.13)	70.40(1.40)	85.00(2.14)
SDONPP	84.73(2.72)	91.04(0.81)	92.96(0.68)	94.96(1.02)	79.32(1.54)	88.50(0.87)	90.41(0.58)	91.57(0.44)
SS-KDA	87.75(2.84)	89.44(1.77)	89.33(0.84)	90.73(1.36)	78.16(3.90)	90.50(0.69)	91.94(1.02)	91.57(1.40)
PCA	85.86(2.55)	92.39(1.11)	93.97(0.80)	95.68(0.85)	80.01(1.57)	88.40(0.72)	90.26(0.53)	91.80(0.73)
Test	α=5	α=20	α=30	α= 40	α=5	α=20	α=30	α=40
Reg-GeoFeature	94.19(2.26)	97.19(0.44)	**97.78(0.24)**	**98.26(0.32)**	**88.44(1.61)**	**91.82(0.80)**	92.30(0.31)	92.30(0.23)
Multi-MDA	**94.96(2.16)**	**97.34(0.44)**	97.67(0.18)	97.70(0.24)	81.68(1.60)	89.84(0.36)	91.09(0.29)	91.96(0.60)
E-Isomap	84.02(3.70)	94.34(1.38)	95.90(0.94)	96.58(0.99)	69.00(4.18)	79.54(2.60)	83.82(0.59)	85.98(2.02)
LDA	87.79(2.67)	85.81(1.92)	85.57(1.57)	84.37(1.81)	83.46(2.52)	73.70(1.53)	76.50(1.48)	84.68(0.75)
IsoProjection	78.50(4.47)	87.36(0.69)	89.36(0.79)	88.05(1.25)	83.37(3.31)	76.09(0.80)	78.50(2.48)	87.26(1.42)
MMP	85.73(1.71)	88.95(0.70)	90.12(0.70)	91.05(0.63)	84.34(2.11)	91.69(0.88)	**92.41(0.48)**	**92.98(0.55)**
LSDA	75.41(1.75)	85.06(1.17)	82.17(2.89)	75.63(5.64)	79.41(2.20)	61.80(2.25)	70.86(3.48)	86.02(2.31)
SDONPP	85.33(2.50)	92.12(0.53)	93.92(0.44)	95.40(0.60)	80.45(1.74)	88.96(0.59)	91.54(0.56)	92.34(0.36)
SS-KDA	85.54(3.04)	89.16(1.66)	89.68(0.87)	91.56(1.40)	74.60(7.60)	87.90(9.52)	89.68(7.14)	90.86(6.81)
PCA	86.24(2.49)	92.80(0.43)	94.46(0.44)	95.72(0.75)	80.70(0.89)	88.97(0.35)	91.14(0.45)	92.22(0.23)

for our datasets, respectively. Second, there are a large number of data points in each class and data of different classes overlap with each other heavily. Thus, separability of the different classes cannot be well characterized by the interclass scatter matrix.

2. Linear methods sometimes can be seen to outperform SS-KDA. This is because the kernel width is very difficult to adapt to a varying number of training points. It is observed that when the number of labeled training points changes, the kernel width needs to be updated accordingly. Otherwise, kernel methods give poor results.
3. If the number of data points in each class is large enough to characterize its data distribution (manifold structure), Reg-GeoFeature and Multi-MDA always show a better performance than the other methods. However, as they are based on manifold assumption and dense data distribution (as is described in the Appendix), Reg-GeoFeature and Multi-MDA are expected to have poor performances when the number of data points in each class is small, such as the face recognition problem when there are only a few images per person.

V. CONCLUSION

In this paper, we have proposed a new multi-manifold learning algorithm. It combines semi-supervised multi-manifold modeling, nonlinear feature extraction, and a new semi-supervised dimension reduction method to achieve a better performance in geodesic feature extraction and classification tasks. Experiments show that the proposed algorithm yields good results on projecting the data to a comparably low-dimensional space. Future work will be concentrated on examining other nonlinear features from data.

APPENDIX A

In the following, we show that f_i in $\mathcal{F}$ uniquely expresses x_i in dataset $\mathcal{X}$ and therefore the replacement of x_i with f_i is appropriate for classification.

Let us first give the definitions of parameterized manifold, tangent map on a manifold and isometry [16].

Definition A.1 (Parameterized Manifold): Let $d \leq D$, and Ω open in $\mathbb{R}^d$. Let $\phi : \Omega \to R^D$. The set $\mathcal{M} \equiv \phi(\Omega)$ together with the mapping ϕ is called a parameterized manifold of dimension d.

Definition A.2 (Tangent Map): The tangent map ϕ_* of ϕ assigns to each tangent vector v of Ω the tangent vector $\phi_*(v)$ of $\mathcal{M}$ such that if v is the initial velocity of a curve γ in Ω, then $\phi_*(v)$ is the initial velocity of the image curve $\phi_*(\gamma)$ in $\mathcal{M}$.

As can be seen, $\mathcal{M}$ is characterized by D functions of d variables, which can be considered as a d-dimensional surface embedded in $\mathbb{R}^D$.

Definition A.3 (Isometry): The mapping $\phi : \Omega \to \mathcal{M}$ is an isometry if ϕ is one-to-one and onto and ϕ preserves inner products in the tangent spaces, i.e., for the tangent map ϕ_*,

$$\phi_*(v)^T \phi_*(w) = v^T w$$

for any two vectors v and w that are tangent to Ω.

For an isometry ϕ defined on an open convex set of $\mathbb{R}^d$, it is easy to show that the geodesic distance between two points $\phi(v)$ and $\phi(w)$ on $\mathcal{M}$ is given by

$$d_G(\phi(v), \phi(w)) = \|v - w\|.$$

The following gives our propositions and results.

Proposition A.1: $\mathcal{M} \equiv \phi(\mathcal{L}) \subset \mathbb{R}^D$ is a d-dimensional parameterized manifold; here $\mathcal{L}$ is an open set in $\mathbb{R}^d$ and $\phi : \mathcal{L} \to \mathbb{R}^D$ is an isometry.

Assume that data points of $\mathcal{X} = \{x_1, \ldots, x_N\}$ densely reside on the manifold $\mathcal{M}$. Correspondingly, we have $\mathcal{Z} = \{z_1, \ldots, z_N\} \subset \mathcal{L}$, where $z_i = \phi^{-1}(x_i) \in \mathbb{R}^d$, $i = 1 \ldots, N$, and ϕ^{-1} denotes the inverse function of ϕ. As coordinate translation is a kind of smooth invertible function which does not involve scale variation and rotations, we assume that ϕ is a compound function involving a translation function. So without loss of generality, we can assume that $\mathcal{Z}$ has a zero mean.

Proposition A.2: $\mathcal{Z} = \{z_1, \ldots, z_N\}$ spans a d-dimensional linear space.

According to Proposition A.2, the dimension of $\mathcal{Z}$ is irreducible. This is a reasonable assumption as $\mathcal{X}$ densely distributes over $\mathcal{X}$, and correspondingly, $\mathcal{Z}$ also densely distributes over $\mathcal{L}$. Let $d_G(x_i, x_j)$ denotes the geodesic distance on $\mathcal{M}$ between data points x_i and x_j. We have $d_G(x_i, x_j) = \|z_i - z_j\|$ as ϕ is an isometry.

For any point $t \in \mathcal{M}$, we define a geodesic distance based feature function f as

$$f(t) = \begin{pmatrix} d_G(x_1, t) \\ \vdots \\ d_G(x_N, t) \end{pmatrix}. \tag{21}$$

Corollary A.1: For any point $t \in \mathcal{M}$, there exists $\theta \in \mathcal{L}$, such that $t = \phi(\theta)$ and

$$f(t) = f \circ \phi(\theta) = \begin{pmatrix} \|z_1 - \theta\| \\ \vdots \\ \|z_N - \theta\| \end{pmatrix} = F(\theta), \tag{22}$$

We see that $F = f \circ \phi$.

As the mapping ϕ is an isometry, this corollary holds.

Theorem A.1: For any $x, y, z \in \mathcal{L}$, if $\|x\| \leq \|y\| \leq \|z\|$, we have $\|F(x)\| \leq \|F(y)\| \leq \|F(z)\|$.

Proof:

$$\|F(\theta)\|^2 = \sum_{i=1}^{N} \|\theta - z_i\|^2 \tag{23}$$

$$= \sum_{i=1}^{N} \left\{ \|z_i\|^2 + \|\theta\|^2 - 2\langle z_i, \theta \rangle \right\} \tag{24}$$

As $\sum_{i=1}^{N} z_i = 0$, thus

$$\|F(\theta)\|^2 = N\|\theta\|^2 + \sum_{i=1}^{N} \|z_i\|^2. \tag{25}$$

Therefore, we have $\|F(x)\| \leq \|F(y)\| \leq \|F(z)\|$ when $\|x\| \leq \|y\| \leq \|z\|$. ■

Here we present the main result:

Theorem A.2: The function $F(\theta)$ is an injection.

Proof: We prove by contradiction: if $\exists p \neq q$, with $p, q \in \mathcal{L}$, we have $F(p) = F(q)$, i.e., $\|z_i - p\| = \|z_i - q\|$, $i = 1, \ldots, N$. Thus, z_i is located on the perpendicular bisector hyper-plane of points p and q.

Let $v_1 = \frac{p-q}{\|p-q\|}$, which can be expanded to a set of basis vectors of space $\mathcal{L}$ by adding $d - 1$ orthonormal vectors$\{v_2, \ldots, v_d\}$. We write $z_i \in \text{span}\{v_2, \ldots, v_d\} + \frac{p+q}{2}$, $i = 1, \ldots, N$.

It can be seen that $\mathcal{Z}$ lies on a $d-1$ linear subspace, which is an contradictory to Proposition A.2. ■

As ϕ is a injection, f is also an injection. Therefore, using feature vectors to represent the original data points makes sense.

REFERENCES

[1] H. Murase and S. K. Nayar, "Visual learning and recognition of 3-D objects from appearance," *Int. J. Comput. Vis.* vol. 14, no. 1, pp. 5–24, Nov. 1995.

[2] P. Belhumeur, J. Hespanha, and D. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 6, pp. 711–720, Jul. 1997.

[3] M. Shafiei *et al.*, "Document representation and dimension reduction for text clustering," in *Proc. IEEE 23rd Int. Conf. Data Eng. Workshop*, Apr. 2007, pp. 770–779.

[4] I. Jolliffe, *Principal Component Analysis*. New York, NY, USA: Springer-Varlag, 1989.

[5] H. Li, T. Jianh, and K. Zhang, "Efficient and robust feature extraction by maximum margin criterion," *IEEE Trans. Neural Netw.*, vol. 17, no. 1, pp. 157–165, Jan. 2006.

[6] J. Tenenbaum, V. Sliva, and J. Landford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, Dec. 2000.

[7] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by local linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.

[8] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Comput.*, vol. 15, no. 6, pp. 1373–1396, Jun. 2003.

[9] D. Donoho and C. Grimes, "Hessian eigenmaps: New locally linear embedding techniques for high-dimensional data," *Proc. Nat. Acad. Sci. United States Amer.*, vol. 100, no. 10, pp. 5591–5596, May 2003.

[10] Z. Y. Zhang and H. Y. Zha, "Principal manifolds and nonlinear dimensionality reduction via tangent space alignment," *SIAM J. Sci. Comput.*, vol. 26, no. 1, pp. 313–338, 2005.

[11] 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.

[12] 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.

[13] T. Zhang, K. Huang, X. Li, J. Yang, and D. Tao, "Discriminative orthogonal neighborhood-preserving projections for classification," *IEEE Trans. Syst., Man, Cybern. Part B, Cybern.*, vol. 40, no. 1, pp. 253–263, Feb. 2010.

[14] S. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 40–51, Jan. 2007.

[15] J. A. Lee, A. Lendasse, and M. Verleysen, "Nonlinear projection with curvilinear distances: Isomap versus curvilinear distance analysis," *Neurocomputing*, vol. 57, pp. 49–76, Mar. 2004.

[16] H. Zha and Z. Zhang, "Continuum Isomap for manifold learning," *Comput. Statist. Data Anal.* vol. 52, no. 1, pp. 184–200, Sep. 2007.

[17] M. Fan, X. Zhang, Z. Lin, Z. Zhang, and H. Bao, "Geodesic based semi-supervised multi-manifold feature extraction," in *Proc. IEEE 12th ICDM*, Dec. 2012, pp. 852–857.

[18] D. Cai, X. He, and J. Han, "Isometric projection," in *Proc. 22nd Conf. Artif. Intell.* Vancouver, Canada, Jul. 2007, pp. 528–533.

[19] D. Cai, X. He, and J. Han, "Learning a maximum margin subspace for image retrieval," *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 2, pp. 189–201, Feb. 2008.

[20] D. Cai, X. He, K. Zhou, J. Han, and H. Bao, "Locality sensitive discriminant analysis," in *Proc. IJCAI* Hyderabad, India, Jan. 2007, pp. 708–713.

[21] S. Yan, D. Xu, Q. Yang, L. Zhang, X. Tang, and H.-J. Zhang, "Multilinear discriminant analysis for face recognition," *IEEE Trans. Image Process.*, vol. 16, no. 1, pp. 212–220, Jan. 2007.

[22] X. Wang and X. Tang, "Dual-space linear discriminant analysis for face recognition," in *Proc. IEEE Conf. CVPR*, Jul. 2004, pp. 564–569.

[23] X. Wang and X. Tang, "A unified framework for subspace face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1222–1228, Sep. 2004.

[24] J. Yang, A. Frangi, J. Yang, D. Zhang, and Z. Jin, "KPCA plus LDA: A complete kernel Fisher discriminant framework for feature extraction and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 2, pp. 230–244, Feb. 2005.

[25] D. Cai, X. He, and J. Han, "Semi-supervised discriminant analysis," in *Proc. IEEE ICCV*, Oct. 2007, pp. 1–7.

[26] Y. Song, F. Nie, C. Zhang, and S. Xiang, "A unified framework for semi-supervised dimensionality reduction," *Pattern Recognit.*, vol. 41, no. 9, pp. 2789–2799, Sep. 2008.

[27] F. Nie, D. Xu, I. W.-H. Tsang, and C. Zhang, "Flexible manifold embedding: A framework for semi-supervised and unsupervised dimension reduction," *IEEE Trans. Image Process.*, vol. 19, no. 7, pp. 1921–1932, Jul. 2010.

[28] W. Yang, C. Sun, and L. Zhang, "A multi-manifold discriminant analysis method for image feature extraction," *Pattern Recognit.* vol. 44, no. 8, pp. 1649–1657, Aug. 2011.

[29] R. Wang and X. Chen, "Manifold discriminant analysis," in *Proc. IEEE Conf. CVPR*, Jun. 2009, pp. 429–436.

[30] H. Chen, H. Chang, and T. Liu, "Local discriminant embedding and its variants," in *Proc. IEEE Conf. CVPR*, Jun. 2005, pp. 846–853.

[31] D. Zhao, Z. Lin, and X. Tang, "Classification via Semi-Riemannian spaces," in *Proc. IEEE Conf. CVPR*, Jun. 2008, pp. 1–8.

[32] W. Zhang, Z. Lin, and X. Tang, "Learning semi-Riemannian metrics for semisupervised feature extraction," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 4, pp. 600–611, Apr. 2011.

[33] J. Lu, Y. Tan, and G. Wang, "Discriminative multimanifold analysis for face recognition from a single training sample per person," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 39–51, Jan. 2013.

[34] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, Nov. 2013.

[35] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013.

[36] Z. Lin, R. Liu, and Z. Su, "Linearized alternating direction method with adaptive penalty for low rank representation," in *Proc. Adv. NIPS*, 2011, pp. 612–620.

[37] X. Geng, D. Zhang, and Z. Zhou, "Supervised nonlinear dimensionality reduction for visualization and classification," *IEEE Trans Syst., Man, Cybern. Part B, Cybern.*, vol. 35, no. 6, pp. 1098–1107, Dec. 2005.

[38] M. Yang, "Extended isomap for pattern classification," in *Proc. 16th Int. Conf. Pattern Recognit.*, 2002, pp. 615–618.

[39] H. S. Seung and D. D. Lee, "The manifold ways of perception," *Science*, vol. 290, pp. 2268–2269, Dec. 2000.

[40] L. Wang and D. Suter, "Visual learning and recognition of sequential data manifolds with applications to human movement analysis," *Comput. Vis. Image Understand.*, vol. 110, no. 2, pp. 153–172, May 2008.

[41] A. Elgammal and C. S. Lee, "Tracking people on a torus," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 3, pp. 520–538, Mar. 2009.

[42] L. Yang, "Building k-connected neighborhood graphs for isometric data embedding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 5, pp. 827–831, May 2006.

[43] M. Balasubramanian, E. L. Schwartz, J. B. Tenenbaum, V. D. Silva, and J. C. Langford, "The Isomap algorithm and topological stability," *Science*, vol. 295, no. 5552, p. 7, Jan. 2002.

[44] N. Gu, M. Fan, H. Qiao, and B. Zhang, "Discriminative sparsity preserving projections for semi-supervised dimensionality reduction," *IEEE Signal Process. Lett.*, vol. 19, no. 7, pp. 391–394, Jul. 2012.

[45] X. Cao, H. Qiao, and J. Keane, "A low-cost pedestrian-detection system with a single optical camera," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 58–67, Mar. 2008.

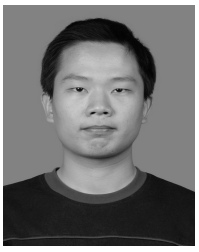
[46] J. Hull, "A database for handwritten text recognition research," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 5, pp. 550–554, May 1998.

[47] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[48] CBCL Face Database, Philadelphia PA, USA. (2001, Jan.). *MIT Center For Biological and Computation Learning* [Online]. Available: http://www.ai.mit.edu/projects/cbcl

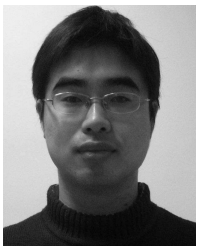
[49] T. Evgeniou, T. Poggio, M. Pontil, and A. Verri, "Regularization and statistical learning theory for data analysis," *Comput. Statist. Data Anal.*, vol. 38, no. 4, pp. 421–432, Feb. 2002.

[50] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *J. Mach. Learn. Res.*, vol. 7, pp. 2399–2434, Nov. 2006.



Mingyu Fan received the B.Sc. degree in mathematics and information science from the Central University for Nationalities, Beijing, China, and the Ph.D. degree in machine learning from the Academy of Mathematics and System Science, Chinese Academy of Sciences, Beijing, in 2006 and 2011, respectively.

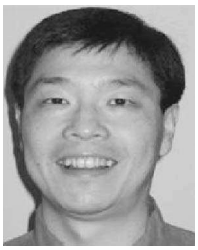
He is currently an Associate Professor with Wenzhou University, Zhejiang, China. His current research interests include feature learning and image classification.



Xiaoqin Zhang (M'12) received the B.Sc. degree in electronic information science and technology from Central South University, Hunan, China, and the Ph.D. degree in pattern recognition and intelligent system from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2005 and 2010, respectively.

He is currently an Associate Professor with Wenzhou University, Zhejiang, China. His current research interests include visual tracking, low-rank recovery, and image classification. He has published more than 30 papers in international and national journals, and international conferences.

Zhouchen Lin received the Ph.D. degree in applied mathematics from Peking University in 2000. He is currently a Professor with the Key Laboratory of Machine Perception, Minister of Education, School of Electronics Engineering and Computer Science, Peking University. He is also a Chair Professor with Northeast Normal University. Before 2012, he was a Lead Researcher with the Visual Computing Group, Microsoft Research Asia. He was a Guest Professor with Shanghai Jiao Tong University, Beijing Jiao Tong University, and Southeast University. He was also a Guest Researcher with the Institute of Computing Technology, Chinese Academy of Sciences. His research interests include computer vision, image processing, computer graphics, machine learning, pattern recognition, and numerical computation and optimization. He is an Associate Editor of *Neurocomputing*.



Zhongfei Zhang received the B.S. degree in electronics engineering and the M.S. degree in information science from Zhejiang University, China, and the Ph.D. degree in computer science from the University of Massachusetts at Amherst.

He is a Professor of Computer Science with the State University of New York at Binghamton. His research interests include computer vision and multimedia processing.

Hujun Bao received the B.S. and Ph.D. degrees in applied mathematics from Zhejiang University in 1987 and 1993, respectively.

He is currently a Professor and the Director of the State Key Laboratory of CAD&CG, Zhejiang University. His main research interests include computer graphics and computer vision, including real-time rendering technique, geometry computing, virtual reality, and structure-from-motion.