Pattern Recognition 76 (2018) 662–678

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Semi-supervised local multi-manifold Isomap by linear embedding for feature extraction

Yan Zhang[a], Zhao Zhang[a,*], Jie Qin[b], Li Zhang[a], Bing Li[c], Fanzhang Li[a]

[a] School of Computer Science and Technology & Provincial Key Laboratory for Computer Information Processing Technology, Soochow University, Suzhou, 215006, PR China
[b] Computer Vision Laboratory, ETH Zürich, 8092, Zürich, Switzerland
[c] School of Economics, Wuhan University of Technology, No.122 Luoshi Road, Wuhan, 430070, PR China

ARTICLE INFO

Article history:
Received 1 October 2016
Revised 29 September 2017
Accepted 30 September 2017
Available online 13 October 2017

Keywords:
Semi-supervised manifold feature extraction
Local multi-manifold Isomap
Linear embedding
Classification

ABSTRACT

In this paper, we mainly propose a semi-supervised local multi-manifold Isomap learning framework by linear embedding, termed SSMM-Isomap, that can apply the labeled and unlabeled training samples to perform the joint learning of neighborhood preserving local nonlinear manifold features and a linear feature extractor. The formulation of SSMM-Isomap aims at minimizing pairwise distances of intra-class points in the same manifold and maximizing the distances over different manifolds. To enhance the performance of nonlinear manifold feature learning, we also incorporate the neighborhood reconstruction error to preserve local topology structures between both labeled and unlabeled samples. To enable our SSMM-Isomap to extract local manifold features from the outside new data, we also add a feature approximation error that correlates manifold features with embedded features by the jointly learnt feature extractor. Thus, the learnt linear extractor can extract the local manifold features from the new data efficiently by direct embedding. To optimize the proposed objective function, two effective schemes are presented, i.e., Scaling by MAjorizing a Complicated Function and Eigen-decomposition. Notice that the comparison of the proposed two solvers is also described. We mainly evaluate SSMM-Isomap for manifold feature learning, data clustering and classification. Extensive simulation results verify the effectiveness of our SSMM-Isomap algorithm, compared with other related feature learning techniques.

1. Introduction

In the fields of pattern recognition and data mining, feature extraction is always a fundamental problem, since it directly affects the performance of the subsequent pattern recognition and data mining models. Note that most real-world data lies in high-dimensional input space, which may result in decreased efficiency and suffer from the "curse of dimensionality" [1,2]. Thus, extracting informative features with lower dimensions from high-dimensional data has been attracting considerable attention during the last decades [48–50]. Generally, feature extraction includes two main tasks, i.e. reducing dimension appropriately and looking for the compact representations [16,26]. The basic principle is to compute a mapping f that can embed each data x in the high-dimensional space $\mathbb{R}^D$ into the compact representation y in a reduced feature space $\mathbb{R}^d$:

$$f : \mathbb{R}^D \to \mathbb{R}^d, \quad x \to y, \quad d \ll D. \tag{1}$$

Broadly speaking, existing feature extraction models can be divided into the unsupervised, supervised and semi-supervised ones according to how much supervised prior knowledge of data is available. Unsupervised learning algorithms only rely on data points without needing to know any supervised prior information for feature learning by explicitly preserving the intrinsic local or global geometrical structures, for instance *Principal Component Analysis* (PCA) [2], *Multi-Dimensional Scaling* (MDS) [4,37], *Locally Linear Embedding* (LLE) [1], *Laplacian Eigenmaps* (LE) [16], *Isometric Mapping* (Isomap) [3], *Locality Preserving Projection* (LPP) [38], *Neighborhood Preserving Embedding* (NPE) [40], and Isoprojection [34]. It should be noticed that LPP, NPE and Isoprojection are linear approximations to previous LE, LLE and Isomap, respectively. Compared with LPP, Isoprojection and NPE that can output an underlying projection matrix to reveal the linear relations of samples, LE, LLE and Isomap mainly focus on discovering the nonlinear manifold structures by reducing the number of dimensionality directly without obtaining an explicit mapping. Specifically, Isomap aims at calculating the low-dimensional manifold features by preserving the pairwise distances, while both LE and LLE focus on discover-

* Corresponding author.
E-mail address: cszzhang@gmail.com (Z. Zhang).

https://doi.org/10.1016/j.patcog.2017.09.043
0031-3203/

ing the low-dimensional subspace by preserving the local geometry structures.

Compared with the unsupervised feature learning methods, supervised models focus on discovering discriminant features for inter-class classification. According to the implementation mechanisms, existing supervised learning approaches can be roughly categorized into two kinds. The first kind of methods are mainly driven by class label information associated with the training samples, among which representative methods are *Linear Discriminant Analysis* (LDA) [39], *Supervised Isomap* (S-Isomap) [6], *Supervised LLE* (SLLE) [5] and *Supervised LPP* (SLPP) [7], etc. Note that S-Isomap, SLLE and SLPP are the supervised versions of Isomap, LLE and LPP, respectively. The other kind of methods is mainly guided by the pairwise Cannot-Link (CL) constraint and Must-Link (ML) constraint [8,9]. Compared with class labels, pairwise constraints (PCs) can sometimes provide more supervised prior information and regulate supervision information flexibly [8]. Typical pairwise constrained manifold learning methods are *Marginal Isomap* (M-Isomap) [8] and the *Multi-Manifold Discriminant Isomap* (MMD-Isomap) [9], etc. Note that although both S-Isomap and M-Isomap are supervised versions of Isomap, M-Isomap tends to offer certain advantages over S-Isomap, because M-Isomap incorporates the local pairwise constraints into Isomap and can compute the shortest path distances over the constrained neighborhood graphs. More specifically, M-Isomap aims at separating inter-class neighborhoods in addition to preserving pairwise distances over intra-class neighborhood as S-Isomap.

Recent MMD-Isomap is a multi-manifold extension of the previous M-Isomap and it enhances M-Isomap by maximizing all inter-class distances in the reduced space. In the setting of MMD-Isomap, it also considers the case that data of different categories lies on different manifolds [9], i.e., *multi-manifold*.

Although promising results are reported, MMD-Isomap still has three obvious drawbacks. First, MMD-Isomap is a fully supervised method, i.e., all the training data need to be labeled prior to manifold feature learning. But it is usually very hard and costly to obtain large amount of label information of samples in practice [10,15,51–53], i.e., class information of samples is often limited and the labeling process is costly. Thus, MMD-Isomap may suffer from the restricted applications as other supervised feature learning methods. Second, MMD-Isomap handles the outside new data by using the back-propagation (BP) neural networks [44] to calculate a deterministic mapping via an independent step after the manifold feature learning process of MMD-Isomap, which is similar to S-Isomap [6]. But on one hand, learning a mapping using an independent process after the manifold feature learning stage cannot explicitly ensure the learnt mapping is optimal for data representation. On the other hand, any nonlinear manifold learning algorithm can use the back-propagation (BP) neural networks or other existing methods to compute the mapping based on the pre-calculated manifold features, but they still suffer from the above issue. Third, MMD-Isomap computes the compact representations by maximizing inter-class distances over different manifolds while preserving pairwise distances on the same manifold directly, i.e., it does not consider to preserve local information of data, which may potentially decrease performance.

In this paper, we therefore propose an enhanced method of MMD-Isomap by addressing its aforementioned shortcomings. The main contributions can be summarized as follows: (1) A *Semi-Supervised Local Multi-Manifold Isomap framework*, called SSMM-Isomap, is technically proposed for manifold feature extraction. Our SSMM-Isomap deals with the case, i.e., partial supervised prior knowledge of samples is available, by jointly performing neighborhood preserving local nonlinear manifold feature learning and linear feature extractor learning. For local nonlinear manifold feature learning, SSMM-Isomap can work effectively using the relationships between labeled and unlabeled data as other existing semi-supervised feature extraction methods [11,12,41–42], such as *Semi-Supervised Linear Discriminant Analysis* (SSLDA) [13], *Semi-Supervised Maximal Margin Criterion* (SSMMC) [14] and *Laplacian LDA* (LapLDA) [33], etc. (2) We propose a learning scheme by linear embedding to extend the nonlinear manifold learning model into the linear scenario for learning linear discriminant features for classification. That is, we incorporate a feature approximation error into existing nonlinear manifold learning framework to correlate manifold features with the embedded features by the learnt linear projection (i.e., feature extractor) so that the learnt projection can handle outside data efficiently by direct embedding. Also, since the projection is jointly learnt and the feature approximation error is jointly minimized, so it can be ensured to be optimal for feature representations. (3) We can consider to keep local neighborhood information in addition to maximizing inter-class distances and preserving those pairwise intra-class distances. Specifically, a LLE-style neighborhood reconstruction error term over reduced features is included into the objective function for joint minimization. (4) Two solution schemes are presented for our algorithm, i.e., *Scaling by MAjorizing a Complicated Function* (SMACOF) [17–19] and by the Eigen-decomposition method [20].

The paper is outlined as follows. Section 2 briefly reviews Isomap and MMD-Isomap. Section 3 proposes our algorithm SSMM-Isomap. Section 4 describes the experimental setting and result. Section V draws the conclusions.

2. Related work

In this section, we briefly review the closely related manifold feature learning methods, i.e., Isomap and MMD-Isomap.

2.1. Isomap

Isomap is one of the most classical global nonlinear manifold learning methods, and aims at seeking an optimal subspace that best preserves the geodesic distance between points [3]. Let$X = (x_1, x_2, ..., x_N) \in \mathbb{R}^{D\times N}$denote a set of N points in the original D-dimensional space, and $Y = (y_1, y_2, ..., y_N) \in \mathbb{R}^{d\times N}$be a set of the reduced representations in the d-dimensional space ($d \ll D$). Then, Isomap can perform manifold feature learning in three steps: 1) Determine the nearest neighbors of each sample by using K-neighborhood or ε-neighborhood [21]; 2) Construct a undirected graph$G(V, E)$, where each node $v_i \in V$corresponds to a pointx_i. $d_G(x_i, x_j)$is the shortest path distances between $x_i$and$x_j$over$G$. Dijkstra's algorithm [23] and Floyd's algorithm [24,25] can be applied to find the shortest paths. Then, Isomap initializes $d_G(x_i, x_j) = d(x_i, x_j)$ supposex$_i$and$x_j$are connected by an edge, and $d_G(x_i, x_j) = +\infty$ otherwise, where $d(x_i, x_j)$is Euclidean distance; 3) Obtain the low-dimensional embedding Y by solving the following problem:

$$\min_{Y} \sum_{i,j} \left(d\left(y_i, y_j\right) - d_G\left(x_i, x_j\right)\right)^2, \tag{2}$$

which can be similarly solved as the classical MDS [4,5,39].

2.2. Multi-manifold discriminant Isomap (MMD-Isomap)

MMD-Isomap is a supervised multi-manifold learning method over Isomap. The main principles of recent MMD-Isomap can be summarized as follows. First, it applies the global pairwise constraints [27–29] to construct the optimization problem. The global pairwise ML and CL sets are defined as

$$ML = \left\{\left(x_i, x_j\right) | i \neq j, l(x_i) = l\left(x_j\right)\right\}, \tag{3}$$

$$CL = \left\{\left(x_i, x_j\right) | i \neq j, l(x_i) \neq l\left(x_j\right)\right\}, \tag{4}$$

where $l(x_i) \in \{1, 2, ..., c\}$ is the class label of $x_i, i = 1, 2, ..., N$, and c is the number of classes. Then, MMD-Isomap aims to preserve the global geometry structures of intra-class data and separate the inter-class data by solving the following two problems:

$$J_{ML} = \min_{Y} \sum_{(x_i,x_j)\in ML} \left(d(y_i, y_j) - d_G(x_i, x_j)\right)^2, \tag{5}$$

$$J_{CL} = \max_{Y} \sum_{(x_i,x_j)\in CL} d^2(y_i, y_j) = \|y_i - y_j\|^2. \tag{6}$$

where $d^2(y_i, y_j)$ denotes the Euclidean distance between the low-dimensional representations y_i and y_j, and $d_G(x_i, x_j)$ is the shortest path distance for approximating the geodesic distance. Minimizing Eq. (5) is equivalent to preserving the pairwise distance between $d(y_i, y_j)$ and $d_G(x_i, x_j)$ for the sample pair $(x_i, x_j) \in ML$, and maximizing Eq. (6) means to separate all the sample pairs $(x_i, x_j) \in CL$. Note that MMD-Isomap constructs one graph for each class when computing geodesic distances, meaning that it constructs c sub-graphs G_l, where $l = 1, 2, ..., c$, and each graph G_l is only constructed using the data belonging to class l. Finally, MMD-Isomap solves the criterion:

$$J = \min_{Y} \sigma(Y),$$

$$where\ \sigma(Y) = \frac{1-\alpha}{|ML|} \sum_{(x_i,x_j)\in ML} \left(d(y_i, y_j) - d_G(x_i, x_j)\right)^2 - \frac{\alpha}{|CL|} \sum_{(x_i,x_j)\in CL} d^2(y_i, y_j), \tag{7}$$

where $\alpha \in (0, 1)$ is a weighting parameter for trading-off the effects of discrimination over the Must-link (ML) and Cannot- Link (CL) constraints, $|ML|$ and $|CL|$ denote the numbers of ML and CL constraints respectively. Note that MMD-Isomap solves Eq. (7) using the Scaling by MAjorizing a COmplicated Function (SMACOF) approach [9].

3. Semi-supervised local multi-manifold Isomap by linear embedding

3.1. Problem formulation

We describe the optimization problem of our SSMM-Isomap that is based on the recent MMD-Isomap, but improves it by extending the manifold feature learning to semi-supervised scenario, linear extension scenario and local feature learning scenario at the same time. Given a set of N training samples in $X = [X_L, X_U] \in \mathbb{R}^{D\times N}$, where D is the original dimensionality of samples, $X_L = [x_1, x_2, ..., x_l] \in \mathbb{R}^{D\times l}$ represents a labeled training set, $X_U = [x_{l+1}, x_{l+2}, ..., x_{l+u}] \in \mathbb{R}^{D\times u}$ is an unlabeled training set, each column $x_i \in \mathbb{R}^D$ denotes a sample, and $l + \mu = N$ denotes the total number of training samples. Different from the previous MMD-Isomap that delivers the low-dimensional embedding directly without outputting an explicit mapping or projection for including new data, considering to preserve neighborhood information and employ both labeled and unlabeled training samples for manifold learning, our SSMM-Isomap can calculate an optimal linear projection $P \in \mathbb{R}^{D\times d}$, where $d << D$, to process outside new data by embedding the new data X into the lower-dimensional representation $Y = [y_1, y_2, ..., y_{l+u}] \in \mathbb{R}^{d\times N}$ and preserve neighborhood information in a semi-supervised manner.

To enable the proposed model to compute low-dimensional local manifold features by using labeled and unlabeled data, a neighborhood preserving regularization $\sum_{i=1}^{N} \|y_i - \sum_{j:x_j\in NN(x_i)} W_{ij} y_j\|_2^2$ over reduced features is clearly incorporated into the problem of MMD-Isomap to build the connection between labeled and unlabeled data by discovering the pairwise local relationships, where $\sum_i W_{ij} = 1$, $NN(x_i)$ denotes the neighbor set of x_i, $\|\bullet\|_2$ is Euclidean distance. To enable the proposed SSMM-Isomap method to learn an explicit projection or feature extractor for handling the outside new data, a feature approximation error $\sum_{i=1}^{N} \|Px_i - y_i\|_2^2$ encoding the mismatch between the embedded features by the extractor and the reduced manifold features is included so that learnt extractor P can embed outside new data efficiently by projection similarly as [22]. Thus, the proposed SSMM-Isomap can preserve local neighborhood information of samples in addition to maximizing those inter-class distances and preserving those pairwise intra-class distances in a semi-supervised manner. These motivate us to define the following objective function:

$$\min_{Y,\ P} \sigma(Y,\ P),\ s.t.\ YY^T = I,\ where$$
$$\sigma(Y,\ P) = \frac{1-\alpha}{|ML|} \sum_{(x_i,x_j)\in ML} \left(d(y_i, y_j) - d_G(x_i, x_j)\right)^2 - \frac{\alpha}{|CL|} \sum_{(x_i,x_j)\in CL} d^2(y_i, y_j) + \beta \sum_{i=1}^{N} \left\| y_i - \sum_{j:x_j\in NN(x_i)} W_{ij} y_j \right\|_2^2, + \gamma \sum_{i=1}^{N} \|Px_i - y_i\|_2^2 \tag{8}$$

where $\alpha \in (0, 1), \beta,\ \gamma$ are the trade-off parameters. Specifically, α mainly trades-off the effects of discrimination over Must-link constraint and Cannot-Link constraint, β trades-off the inter-class discrimination and neighborhood preservation via $\sum_{i=1}^{N} \|y_i - \sum_{j:x_j\in NN(x_i)} W_{ij} y_j\|_2^2$ over labeled and unlabeled data, and γ mainly trades-off the neighborhood preserving discriminative manifold feature learning (i.e., first three terms) and the linear feature approximation via $\sum_{i=1}^{N} \|Px_i - y_i\|_2^2$ for handling outside new data. It should be noted that the objective function of existing MMD-Isomap [9] only has two terms over Must-link and Cannot-Link constraints, so the choice of α is relatively easy, because a large α means that the inter-class separation under the Cannot-Link constraint will dominate the objective function value than intra-class compactness under Must-link constraint, and else the intra-class compactness will dominate the result. Compared with the previous MMD-Isomap method, our model involves two more terms to endow MMD-Isomap the power to preserve neighborhood information in addition to performing inter-class discrimination and also enable existing MMD-Isomap to handle outside new data efficiently by direct embedding, thus the proper choices of trading-off parameters (i.e., α, β,γ) would be difficult in our proposed formulation than MMD-Isomap. Regarding the choice of the parameters, we must say that the optimal determination of those model parameters still remains an open issue in reality, but note that we can have the following underlying discussions based on the term that the parameter is associated in our formulation. First, since $\alpha \in (0, 1)$, if β is set to a large value, the effects of the neighborhood preservation over the labeled and unlabeled data would have a greater impact on the objective function value than the effects of inter-class discrimination. Second, the computation of the feature extractor P only depends on the low-dimensional nonlinear manifold features Y. Thus, suppose β is also set to a small value less than 1 similarly as α, in such cases suppose that γ is set to a very large value, the effects of linear feature approximation would have a greater impact on the objective function value than the effects of discriminative nonlinear manifold learning, and else the objective function value would be determined by trading-off the involved several terms jointly in the formulation. Note that the reconstruction weights matrix W of the training samples can be computed by minimizing the following LLE-style optimization problem:

$$\varepsilon(W) = \min_{W} \sum_{i} \left\| x_i - \sum_{j:x_j\in NN(x_i)} W_{ij} x_j \right\|_2^2. \tag{9}$$

According to the terms J_{ML} and J_{CL} defined in Eqs. (5) and (6), the objective function of our SSMM-Isomap formulation can be rewritten in matrix form as follows:

$$\begin{aligned}&\min_{Y,\,P}\sigma(Y,\,P),\ s.t.\ YY^T=I,\ where\\&\sigma(Y,\,P)=\tfrac{1-\alpha}{|ML|}J_{ML}-\tfrac{\alpha}{|CL|}J_{CL}+\beta\left\|Y-YW^T\right\|_F^2+\gamma\|PX-Y\|_F^2.\end{aligned}\tag{10}$$

Note that for semi-supervised feature learning, controlling the K-neighbors of each sample within the same category as much as possible is important for enhancing the representation. To ensure the K-neighbors of each sample can belonging to the same category as much as possible, our SSMM-Isomap makes the efforts in threefold. First, our SSMM-Isomap involves the Cannot-Link (CL) and Must-Link (ML) constraints defined over class information of the small number of labeled data so that we can obtain more supervised information. Second, the pairwise distances between the sample pairs under the ML constraint (i.e., same manifold) are minimized so that the labeled data of the same class can be pushed close, while the distances between the sample pairs under the CL constraint (i.e., different manifolds) are maximized so that the labeled data from different classes can be separated as much as possible. Third, a neighborhood preserving term $\|Y-YW^T\|_F^2$ is included to preserve the local information of all samples, including both labeled and unlabeled data. As a result, the K-neighbors of each sample in our proposed SSMM-Isomap algorithm can be controlled within the same class as much as possible, which has been verified by the extensive visualization and quantitative clustering evaluation results, that is, high intra-class compactness degree and high inter-class separation degree can be obtained at the same time.

In what follows, we present two effective solution schemes based on the *Scaling by MAjorizing a Complicated Function* (SMACOF) [17–19] and *Eigen-decomposition* [20]. Note that SMACOF solves the problem by iterative optimization, while the Eigen-decomposition approach addresses the problem by performing Eigen-decomposition over a matrix directly.

3.2. Effective solution using SMACOF

We first describe the optimization method by SMACOF [9]. Next, we only show the main procedures of using SMACOF to solve our SSMM-Isomap and the detailed procedures are shown in Appendix B. The main objective of SMACOF is to construct a simpler and more manageable auxiliary function. So we can rewrite the criterion of our SSMM-Isomap as

$$\begin{aligned}J&=\min_{Y,P}\sigma(Y,P)\\\sigma(Y,P)&=\frac{1-\alpha}{|ML|}J_{ML}-\frac{\alpha}{|CL|}J_{CL}+\beta\left\|Y-YW^T\right\|_F^2+\gamma\|PX-Y\|_F^2\\&=\eta^2{}_{con}+\eta^2(Y)-2\rho(Y)+\omega(Y)+\varphi(Y,P),\end{aligned}\tag{11}$$

where each term in the above problem can be expressed as

$$\eta^2{}_{con}=\frac{1-\alpha}{|ML|}\sum_{(x_i,x_j)\in ML}d_G^2(x_i,x_j)\tag{12}$$

$$\eta^2(Y)=\frac{1-\alpha}{|ML|}\sum_{(x_i,x_j)\in ML}d^2(y_i,y_j)-\frac{\alpha}{|CL|}\sum_{(x_i,x_j)\in CL}d^2(y_i,y_j)\tag{13}$$

$$\rho(Y)=\frac{1-\alpha}{|ML|}\sum_{(x_i,x_j)\in ML}d(y_i,y_j)d_G(x_i,x_j)\tag{14}$$

$$\omega(Y)=\beta\left\|Y-YW^T\right\|_F^2=\beta tr\left(\left(Y-YW^T\right)\left(Y-YW^T\right)^T\right)\tag{15}$$

$$\varphi(Y,P)=\gamma\|PX-Y\|_F^2=\gamma tr\left((PX-Y)(PX-Y)^T\right),\tag{16}$$

where Eq. (12) is a constant term. The extractor P in Eq. (16) can be obtained by setting the derivative *w.r.t.* P to zero:

$$\begin{aligned}&\partial\left(\gamma(PX-Y)(PX-Y)^T\right)/\partial P\\&\quad=\partial\left(\gamma\left(PXX^TP^T-PXY^T-YX^TP^T+YY^T\right)\right)/\partial P=0\\&\quad\Rightarrow P=YX^T\left(XX^T\right)^{-1}.\end{aligned}\tag{17}$$

For convenience, we define $V=\sum_{i,j=1,i\neq j}^{N}w_{ij}^{ML}A^{ij}-\sum_{i,j=1,i\neq j}^{N}w_{ij}^{CL}A^{ij}$, $B=\sum_{i,j=1,i\neq j}^{N}w_{ij}A^{ij}$, $M=\beta(I-W^T)(I-W)$ and $K=\gamma(I-X^T(XX^T)^{-1}X)$. After solving the Eqs. (13)–(16) individually, the optimization problem in Eq. (11) can be transformed into

$$\begin{aligned}&\min_{Y,P}\sigma(Y,P),\ where\\&\sigma(Y,P)=\eta^2{}_{con}+\eta^2(Y)-2\rho(Y)+\omega(Y)+\varphi(Y,P)\\&\quad\le\eta^2{}_{con}+tr\left(YVY^T\right)-2tr\left(YB(Z)Z^T\right)\\&\quad+tr\left(YMY^T\right)+tr\left(YKY^T\right)\\&\quad=\eta^2{}_{con}+tr\left(Y(V+M+K)Y^T\right)-2tr\left(YB(Z)Z^T\right)\\&\quad=\tau(Y,Z),\end{aligned}\tag{18}$$

where $\tau(Y,Z)=\eta^2{}_{con}+tr(Y(V+M+K)Y^T)-2tr(YB(Z)Z^T)$ denotes an auxiliary function. Then, low-dimensional features Y can be obtained by setting the derivative of Eq. (18) *w.r.t.* Y to 0:

$$\begin{aligned}&\partial\tau(Y,Z)/\partial Y=2Y(V+M+K)-2ZB(Z)=0\\&\qquad\Rightarrow Y=ZB(Z)(V+M+K)^{-1}.\end{aligned}\tag{19}$$

We summarize the complete SMACOF based optimization procedures for our SSMM-Isomap formulation in Table 1. Note that the process of SMACOF majorization is shown in Appendix A. To evaluate the convergence of our model, we present the convergence analysis of the objective function in Fig. 1 on YALE face database [30], JAFFE face database [31] and COIL-20 object database [32] as examples, where the X axis represents the number of iterations and the corresponding Y axis is the difference between two objective function values. In this study, we use 6 images with 3 labeled from each class of the YALE face database, 10 images with 6 labeled from each class of the JAFFE face database, and 35 images with 20 labeled from each object of the COIL-20 object database for training in our proposed SSMM-Isomap, respectively. We can see from the convergence curves in Fig. 1 that the convergence speed of our SSMM-Isomap algorithm is very fast.

3.3. Effective solution using Eigen-decomposition

In addition to solving the problem of our SSMM-Isomap by using iterative SMACOF approach, we also provide another solution scheme by using Eigen-decomposition. To facilitate the optimization, we firstly transform the four terms in Eq. (10) into the following trace form based expressions:

$$\frac{1-\alpha}{|ML|}J_{ML}=\frac{1-\alpha}{|ML|}\sum_{(x_i,x_j)\in ML}\left(d(y_i,y_j)-d_G(x_i,x_j)\right)^2=tr\left(YR^{ML}Y^T\right),\tag{20}$$

Table 1
SSMM-Isomap algorithm by SMACOF.

Inputs: Data matrix $X \in \mathbb{R}^{D\times N}$, label set $L \in \mathbb{R}^{1\times N}$, neighborhood size K, parameters α, β, γ.
Preparation:
1. Construct the *ML* and *CL* sets using Eqs. (3) and (4);
2. Search the *K*-neighbors of each sample;
3. Compute the shortest path distance $d_G(x_i, x_j)$ of each $(x_i, x_j) \in ML$ using Dijkstra's or Floyd's algorithm;
Optimization process:
1. Initialize $Z = Y^0$, where $Y^0 \in \mathbb{R}^{d\times N}$ is a random matrix; obtain w_{ij}^{ML} and w_{ij}^{CL} by Eq. (33);
2. Calculate the reconstruction coefficient matrix W using Eq. (9);
3. Construct the matrix V by Eq. (35) and compute V^{-1}; calculate $\sigma(Y^0)$ by Eq. (11); set $t = 0$;
4. Update P, w_{ij} respectively using Eqs. (17), (37) and compute $B = \sum_{i,j=1,i\neq j}^{N} w_{ij}A^{ij}$;
5. Compute $Y^{t+1} = ZB(V + M + K)^{-1}$;
6. Compute $\sigma(Y^{t+1})$ by Eq. (11); if $\sigma(Y^t) - \sigma(Y^{t+1}) < \xi$, stop; else let $Z = Y^{t+1}$, $t = t + 1$ and go to Step 4.
Outputs: Optimal embedding Y^* and linear projection P^*.

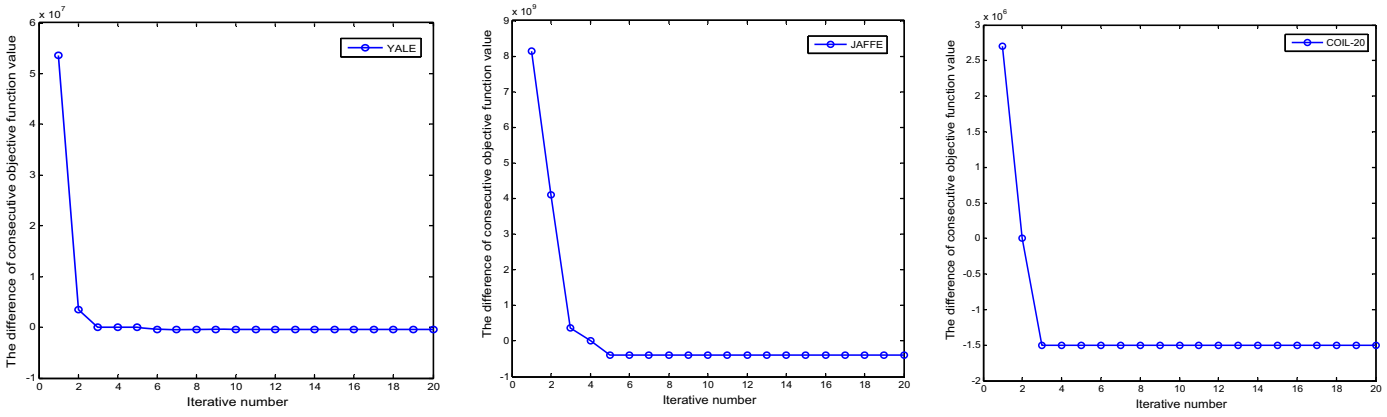


Fig. 1. Convergence analysis of our SSMM-Isomap by SMACOF on the YALE, JAFFE and COIL-20 image databases.

$$\frac{\alpha}{|CL|}J_{CL} = \frac{\alpha}{|CL|}\sum_{(x_i,x_j)\in CL} d^2(y_i, y_j) = \sum_{i,j=1,i\neq j}^{N} w_{ij}^{CL} d^2(y_i, y_j)$$
$$= tr\left(Y\left(\sum_{i,j=1,i\neq j}^{N} w_{ij}^{CL}A^{ij}\right)Y^T\right) = tr\left(YV_{CL}Y^T\right), \quad (21)$$

$$\beta\left\|Y - YW^T\right\|_F^2 = \beta\left\|Y\left(I - W^T\right)\right\|_F^2 = tr\left(YMY^T\right), \quad (22)$$

$$\gamma\|PX - Y\|_F^2 = \gamma\left\|YX^T\left(XX^T\right)^{-1}X - Y\right\|_F^2 = tr\left(YKY^T\right), \quad (23)$$

where $R^{ML} = -(1-\alpha)(H(Q^{ML})^2H)/2|ML|$, $Q_{ij}^{ML} = (d_G^{ML}(x_i, x_j))^2$, $V_{CL} = \sum_{i,j=1,i\neq j}^{N} w_{ij}^{CL}A^{ij}$, and $H = I - ee^T/N$ is a centering matrix and e denotes a vector of all ones. The fourth equation holds since $P = YX^T(XX^T)^{-1}$ in Eq. (17). Based on substituting the above equations back into Eq. (10), the optimization problem in Eq. (10) can be rewritten as

$$\max_Y tr\left(Y\left(R^{ML} + V_{CL} - M - K\right)Y^T\right), s.t.\ YY^T = I. \quad (24)$$

Let $A = R^{ML} + V_{CL} - M - K$, we can obtain the d-dimensional embedding Y as $[\sqrt{\lambda_1}v_1, \sqrt{\lambda_2}v_2, ..., \sqrt{\lambda_d}v_d]^T$, where v_p, $p = 1, 2,...,d$, are the standard eigenvectors corresponding to first d leading eigenvalues (in decreasing order) of the matrix A. After the d-dimensional embedding Y is obtained, the linear projection P can be similarly obtained by Eq. (17). It is worth noting that the optimization procedures of our SSMM-Isomap by using Eigen-decomposition are summarized in Table 2.

3.4. Computational time complexity

In this paper, we have mainly presented two solution schemes (i.e., by the iterative SMACOF and Eigen-decomposition) to optimize our proposed objective function, so we would like to present the complexity analysis for the proposed two solution schemes. For the first scheme by using the iterative SMACOF, solving the problem of our SSMM-Isomap needs to compute multiplication and inverse operations of matrix$X \in \mathbb{R}^{D\times N}$ iteratively, resembling the MMD-Isomap [9], so this solution scheme has a time complexity of $O(tdN^2 + N^3)$, where t is the number of iterations and d is the reduced dimension. Note that the time complexity of previous MMD-Isomap by SMACOF is the same as that of our algorithm by SMACOF. While the other solution scheme by Eigen-decomposition does not need the iterative computations, since it only requires to obtain the eigenvalues and eigenvectors of$X \in \mathbb{R}^{D\times N}$. So, the solution scheme by applying Eigen-decomposition has a time complexity of $O(N^3)$, where N is the number of samples.

It is clear that the computational cost of the proposed solver by the iterative SMACOF depends more on the number of data points than the number of dimensionality, while the proposed solver by Eigen-decomposition only depends on the number of data points (N). It is also worth noting that the computational complexities of the two solution schemes are the same unless the case that td is functionally growing strictly faster than a linear function of N is satisfied. Although the above case is not discussed in this work, the case can be satisfied in certain real-world applications with the small sample size issue, i.e., the dimensionality of the samples is far larger than the number of data points. For example, given a dataset of 100 images and each image has 200*200 pixels, the dimensionality of the vectorized representation of each image is 40,000. For image feature extraction for the above case, we can theoretically reduce the number of dimensionality (i.e., d) from 1 to 40,000, such as $d = 5000$. Thus, the value of td can grow much faster than N. In this case, td will play a more important role in determining the time complexity than N, i.e., the complexity of $O(tdN^2 + N^3)$ is larger than that of $O(N^3)$. In other words, the ap-

Table 2
SSMM-Isomap algorithm by using Eigen-decomposition.

Inputs: Data matrix $X \in \mathbb{R}^{D\times N}$, label matrix $L \in \mathbb{R}^{l\times N}$, neighborhood size K, parameters α, β, γ.
Preparation:
1. Construct the ML and CL sets using Eqs. (3) and (4);
2. Search the K-neighbors of each sample;
Compute the shortest path distance $d_G(x_i, x_j)$ of each $(x_i, x_j) \in ML$ using Dijkstra's or Floyd's algorithm;
Optimization process:
1. Initialize e is a vector of all ones and I is an identity matrix of size $N \times N$;
2. Compute $H = I - ee^T/N$, $Q_{ij}^{ML} = (d_G^{ML}(x_i, x_j))^2$;
3. Compute R^{ML} by $R^{ML} = -(1-\alpha)(H(Q^{ML})^2H)/2|ML|$;
4. Compute V_{CL} by Eq. (34);
5. Compute M by $M = \beta(I - W^T)(I - W)$ and calculate K by $K = \gamma(I - X^T(XX^T)^{-1}X)$;
6. Compute $A = R^{ML} + V_{CL} - M - K$;
7. Obtain the first d leading eigenvalues (in decreasing order) of the matrix A by feature decomposition;
8. Obtain Y as $[\sqrt{\lambda_1}v_1, \sqrt{\lambda_2}v_2, ..., \sqrt{\lambda_d}v_d]^T$ and obtain P by Eq. (17).
Outputs: Optimal embedding Y^* and linear projection P^*.

proach by using Eigen-decomposition is more efficient than the one by using the iterative SMACOF approach based on the above case. In conclusion, the solver by Eigen-decomposition is more suitable for dealing with the datasets with high-dimensional features and relatively small-scale sizes, while the other solver by iterative SMACOF is more suitable for handling the cases with relatively small numbers of dimensions and samples.

4. Relationship analysis

In this section, we mainly discuss the important issues related to our proposed SSMM-Isomap algorithm.

4.1. Relation to Isomap

Isomap formulates the low-dimensional embedding by solving the following optimization problem:

$$\min_{Y} \sum_{i,j} \left(d(y_i, y_j) - d_G(x_i, x_j)\right)^2, \ s.t.\ YY^T = I. \tag{25}$$

Note that the term constrained by ML set in the criterion of SSMM-Isomap looks like the above problem. Specifically, the ML constrained problem in SSMM-Isomap can be defined as

$$\min_{Y} \sum_{(x_i,x_j)\in ML} \left(d(y_i, y_j) - d_G(x_i, x_j)\right)^2, \ s.t.\ YY^T = I. \tag{26}$$

The difference between the above two problems is that Isomap solves the problem by determining the neighborhoods of each sample with nearest-neighbor search and preserve the pairwise distances between all the samples over a weighted neighborhood graph, while our SSMM-Isomap only considers to preserve pairwise distances between samples in ML set. In other words, the formulation in Eq. (10) can be regarded as the extended version of the formulation in Eq. (26) .

4.2. Relation to MMD-Isomap

Recalling the criterion of SSMM-Isomap in Eq. (10), suppose that β=γ=0, then the problem can be transformed into

$$\min_{Y,P} \ \frac{1-\alpha}{|ML|}J_{ML} - \frac{\alpha}{|CL|}J_{CL}, \ s.t.\ YY^T = I, \tag{27}$$

which is just the objective function of recent MMD-Isomap. That is, MMD-Isomap can be considered as a special example of our proposed SSMM-Isomap formulation.

4.3. Relation to LLE/NPE

To solve the problem of SSMM-Isomap, there is an important step to construct a reconstruction weight matrix W over both labeled and unlabeled samples so that the locality topology information can be effectively preserved. The reconstruction weight matrix W is solved from the following problem:

$$\min_{W} \sum_{i} \left\| x_i - \sum\nolimits_{j:x_j\in NN(x_i)} W_{ij}x_j \right\|_2^2, \ S.t.\ \sum\nolimits_i W_{ij} = 1, \tag{28}$$

where $NN(x_i)$ is the neighbor set of x_i, $\|a-b\|_2$ is Euclidean distance between two vectors a and b. The above weighting method is named the LLE-style reconstruction weights. By replacing the high-dimensional x_i with the corresponding y_i in low-dimensional space, i.e., Eq. (28) can be converted into

$$\min_{Y} \sum_{i} \left\| y_i - \sum\nolimits_{j:x_j\in NN(x_i)} W_{ij}y_j \right\|_2^2, \tag{29}$$

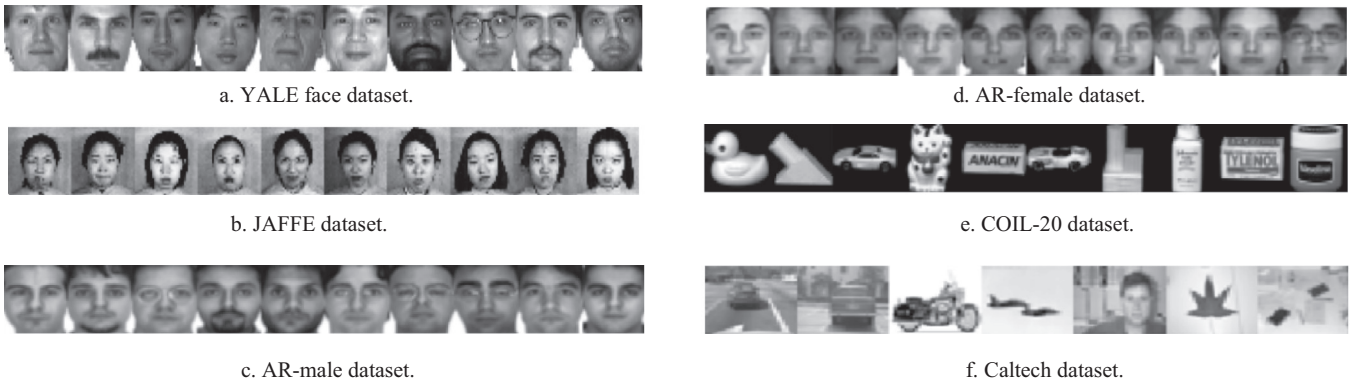
which is just the objective function of LLE. Considering the extreme case that $PX = Y$ in the minimization problem of our SSMM-Isomap, one can obtain a reduced formulation:

$$\min_{Y,P} \ \frac{1-\alpha}{|ML|}J_{ML} - \frac{\alpha}{|CL|}J_{CL} + \beta\left\|PX - PXW^T\right\|_F^2. \tag{30}$$

It should be noted that when label information of all the samples is not available, i.e., the ML and CL sets will not be constructed due to lack of label information. In such case, the above problem can be further reduced to $\min_P \|PX - PXW^T\|_F^2$ subject to $PXX^TP^T = I$, which is just the objective function of NPE. Note that similar argument exists for LLE. That is, both LLE and NPE can also be treated as the special examples of our proposed SSMM-Isomap formulation. Thus, our proposed SSMM-Isomap can potentially outperform existing Isomap, MMD-Isomap, LLE and NPE methods for feature extraction.

5. Experimental results and analysis

In this section, we mainly conduct experiments on several real databases to examine the performance of our SSMM-Isomap method by data clustering and classification. The results are compared with those of several related algorithms, including the unsupervised PCA, Isomap and Isoprojection (IsoP) [34] feature extraction methods, supervised LDA, LapLDA and MMD-Isomap models as well as the semi-supervised SSLDA and SSMMC methods. IsoP is the linear version of Isomap. In the clustering experiments, three real-world datasets, i.e., YALE face database [30], Japanese Female Facial Expression (JAFFE) database [31], and COIL-20 object database [32] are tested. For the classification experiments, we



a. YALE face dataset.
b. JAFFE dataset.
c. AR-male dataset.
d. AR-female dataset.
e. COIL-20 dataset.
f. Caltech dataset.

Fig. 2. Sample images from the evaluated real-world image databases.

Table 3
List of used datasets and dataset information.

Data Type	Dataset Name	#Images (N)	#Class (C)
Face image databases	YALE	165	15
	JAFFE	213	10
	AR-male	1300	50
	AR-female	1300	50
Object image databases	COIL-20	1440	20
	Caltech	3738	7

describe the face and object recognition results. For face recognition, the YALE, JAFFE, AR male database and AR female database [43] are evaluated, while for object recognition, COIL-20 and Caltech object databases (available at http://www.vision.caltech.edu/html-files/) are evaluated. The detailed descriptions of the used databases are described in Table 3. Some sample images of the evaluated databases are shown in Fig. 2.

In all simulations presented in this paper, PCA is used to remove null space or redundant information and for the consideration of efficiency by reducing the dimensions of original data to 200. Note that our model and other compared methods are all performed based on the same reduced feature subspace for a fair comparison. For quantitative evaluations, numerical clustering and classification results are reported. More specifically, the visual and clustering results are shown in Subsection *A*, the face recognition results are illustrated in Subsection *B*, and the object recognition results are illustrated in the Subsection *C*. We also present the parameter sensitivity analysis results in Subsection *D*.

5.1. Visual and quantitative clustering evaluations

We evaluate the clustering performance of our SSMM-Isomap by comparing the visual results and numerical clustering results with several other related dimensionality reduction criteria, i.e., PCA, LDA, Isomap, IsoP, LapLDA, SSLDA, SSMMC and MMD-Isomap. In the visualization and clustering experiments on the COIL-20 database, we use 16 samples of each class with 8 labeled and 8 unlabeled samples as the training set for learning the low-dimension features. In the experiments on the JAFFE and YALE face databases, we choose 6 samples of each class with 3 labeled and 3 unlabeled as training set. It is important to note that the fully-supervised LDA and MMD-Isomap can only apply the labeled training images for learning. Both unsupervised PCA, Isomap, IsoP and the semi-supervised SSMM-Isomap, SSMMC, SSLDA, LapLDA can use both labeled and unlabeled training images for learning. But different from the semi-supervised methods, label information of the labeled data cannot be applied by the unsupervised algorithms.

Note that the numerical clustering evaluation is performed based on the low-dimensional embedding Y of all samples in the databases. Because PCA, LDA, IsoP, LapLDA, SSLDA, SSMMC and our SSMM-Isomap are all linearized methods, i.e., they can deliver an explicit projection that can be used to include the outside new data. As a result, these linear methods can compute a projection P based on the training set to embed all data in given datasets into low-dimensional representations Y. But both Isomap and MMD-Isomap are nonlinear methods, i.e., they cannot output a projection for the inclusion of new data. Thus for fair comparison to PCA, LDA, IsoP, LapLDA, SSLDA, SSMMC and our SSMM-Isomap, we adopt a similar method for Isomap and MMD-Isomap. That is, we involve a separable minimization over the feature approximation error $\|PX_{tr} - Y_{tr}\|_F^2$ after obtaining the low-dimensional embedding Y_{tr} of the training set X_{tr} so that a linear projection P can be learnt. After the linear projection P is obtained by Eq. (17), visualization and the clustering evaluations can be similarly performed as other linear methods.

5.1.1. Visualization results

We present a group of visualization experiments over the COIL-20 database as an example. In this study, we choose seven object classes for simulations for clear observation, where each object class has 72 images. After obtaining the linear projection P, low-dimensional embedding Y can be obtained by embedding all the data onto P. Then, we illustrate the low-dimensional embedding Y in a 2D space (i.e., the number of reduced dimension $d = 2$) in Fig. 3, where the horizontal axis is the first dimension of Y and the vertical axis denotes the second dimension of Y. We can easily find from the visualization results in Fig. 3 that: 1) Compared with other methods, our SSMM-Isomap can provide a clear separation of data points from different manifolds, and obtain the enhanced compactness for intra-class points. Also, our SSMM-Isomap can preserve the neighborhood relations between data points clearly due to the imposed neighborhood reconstruction error term for joint minimization; 2) One can also see that LDA, SSLDA and MMD-Isomap also achieve better results due to the fact that they can make use of available label information of images so that the performance of inter-class separation can be enhanced; 3) The result of IsoP is also good for intra-class compactness due to the capability of preserving the pairwise similarities between data points, but those inter-class points cannot be separated clearly due to the fact that it is essentially an unsupervised feature learning method.

5.1.2. Quantitative clustering evaluation

In addition to visually evaluating the visualization results, we also present the numerical results using *F-measure* and *Normalized Mutual Information* (NMI) [24,35,36,54], which are the most commonly used clustering evaluation metrics. The F-measure is defined

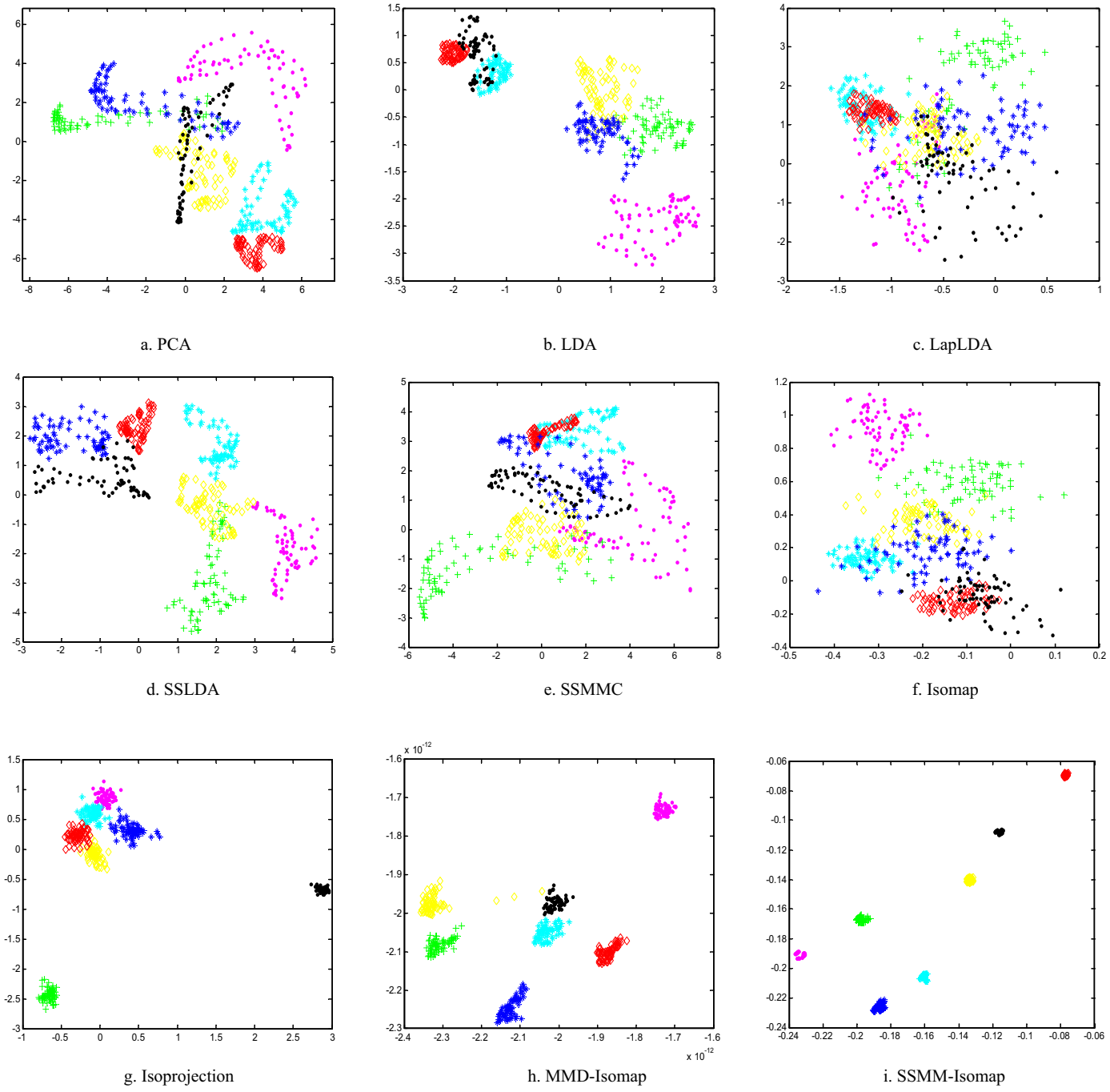


Fig. 3. The two-dimensional (2D) embedding result obtained by each method on the COIL-20 object image database.

as follows:

$$F_\mu = \frac{(\mu^2+1)\text{Precision} \times \text{Recall}}{\mu^2 \text{Precision} + \text{Recall}}, \tag{31}$$

In our experiments, we set parameter $\mu = 1$. Let C be the set of clusters obtained from the ground truth and C_{new} be the set of clusters obtained by performing unsupervised clustering over the features of each algorithm, then the value of *Normalized Mutual Information* (NMI) between C and C_{new} is defined as

$$NMI(C, C_{new}) = \frac{MI(C, C_{new})}{\max(H(C), H(C_{new}))}, \tag{32}$$

where $MI(C,C_{new})$ is the original *Mutual Information* (MI) between C and C_{new}, and $H(C)$ and $H(C_{new})$ are the entropies of C and C_{new} [24,35,36,54]. The values of the *F-measure* and *NMI* range from 0 to 1, that is, the higher the value is, the better the corresponding clustering result will be.

In this study for quantitative evaluations, three real image databases, i.e., JAFFE, YALE and COIL-20, are examined. For clustering analysis, we perform k-means clustering over the low-dimensional representation Y by each method and set the cluster number k to the number of classes in each image database in k-means clustering process. Note that the k-means clustering is easily affected by different initializations of the cluster centroids. Thus, k-means clustering on the learnt low-dimensional features is performed for each method over 30 times to alleviate the aforementioned problem.

Table 4
Quantitative clustering evaluation results.

Methods	JAFFE face database		YALE face database		COIL-20 object database	
	F-measure	*NMI*	*F-measure*	*NMI*	*F-measure*	*NMI*
PCA	0.6077	0.7416 ± 0.0180	0.3238	0.4483 ± 0.0122	0.4552	0.6423 ± 0.0075
LDA	0.6523	0.7700 ± 0.0189	0.4728	0.5157 ± 0.0134	0.4395	0.6518 ± 0.0100
LapLDA	0.4917	0.6301 ± 0.0248	0.3433	0.4652 ± 0.0102	0.2330	0.4140 ± 0.0074
SSLDA	0.7366	0.8410 ± 0.0337	0.3993	0.5394 ± 0.0092	0.4231	0.6023 ± 0.0058
SSMMC	0.3418	0.4833 ± 0.0115	0.3029	0.4822 ± 0.0138	0.2983	0.4869 ± 0.0084
Isomap	0.1979	0.3335 ± 0.0202	0.4329	0.5269 ± 0.0165	0.1068	0.2316 ± 0.0098
IsoP	0.6363	0.7498 ± 0.0102	0.6170	0.7755 ± 0.0312	0.6033	0.7687 ± 0.0084
MMD-Isomap	0.7037	0.8111 ± 0.0349	0.5763	0.7382 ± 0.0171	0.8339	0.9321 ± 0.0180
SSMM-Isomap	**0.7942**	$\mathbf{0.9043 \pm 0.0343}$	**0.6202**	$\mathbf{0.8202 \pm 0.0341}$	**0.8428**	$\mathbf{0.9499 \pm 0.0261}$

We show the comparison results of clustering evaluations with other related methods in Table 4. From the results, we can draw the following conclusions: (1) The quantitative results generally keep consistent with the visualization results in terms of intra-cluster compactness and inter-cluster separation, i.e., our SSMM-Isomap can deliver higher values of NMI and F-measure than other methods. LDA, SSLDA and MMD-Isomap obtain comparable and even better results than other remaining methods. Specifically, MMD-Isomap works better on the COIL-20 object database, because its clustering evaluation results are close to those of our SSMM-Isomap, and both are superior to other methods; (2) The values of NMI delivered by each algorithm are generally higher than the corresponding values of F-measure. The clustering evaluation result of Isomap is usually the worst one in most cases, since it cannot obtain enhanced inter-class separation and intra-class compactness due to its modeling and unsupervised nature.

5.2. Face recognition

In this subsection, we evaluate our SSMM-Isomap and other related methods for representing and recognizing face images over several widely-used benchmark face image databases. The involved face databases include YALE, JAFFE, AR-male and AR-female face databases. Note that our SSMM-Isomap solves the feature extraction and the out-of-sample extension issues by incorporating a feature approximation error term and computing an explicit projection jointly from one objective function so that the learnt projection can handle outside new data efficiently through direct embedding instead of involving the other independent step for computing a mapping after the feature learning process as MMD-Isomap does, thus the face recognition performance of our SSMM-Isomap algorithm is mainly compared with those of linear projection based PCA, LDA, LapLDA, SSLDA, SSMMC and IsoP algorithms. For quantitative recognition evaluations, the widely applied one-nearest-neighbor (1NN) classifier is employed due to its simplicity. For each experimental setting, the results are averaged over 15 random splits of training/test images to avoid the bias.

In this simulation, we evaluate the recognition performance under different numbers of reduced dimensions and training images and illustrate the test results in Figs. 4–7 In the results, SSMM-Isomap1 denotes our method solved by SMACOF and SSMM-Isomap2 denotes our method solved by using Eigen- decomposition. For each fixed number of training images in face database, we vary the number of reduced dimensions and observe the performance trends. From the results in Figs. 4–7, we can have the following findings. First, the face recognition performance of each method can be enhanced by increasing the numbers of reduced dimensions and training face images in virtually all cases. Moreover, for fixed number of training images to evaluate the reduced dimensions, the performance of each algorithm firstly goes up faster when the number of reduced dimensions is relatively small, while the recognition result increases slower as the number of reduced dimensions is increased to a higher level; Second, our SSMM-Isomap methods outperforms others in most cases, especially on the YALE and AR databases. It is also worth noting that SSMM-Isomap2 obtains better results than SSMM-Isomap1 on YALE and AR-female databases, while on the contrary SSMM-Isomap1 beats SSMM-Isomap2 over the JAFFE and AR-male face datasets; Third, one can also see that the overall performance of supervised and semi-supervised methods are more promising than unsupervised methods due to the use of label information.

Note that the statistics according to Figs. 4–7 are shown in Tables 5–8 respectively, where we describe the mean accuracies accompanied with standard deviation (std), and the highest accuracies of evaluated methods. From the statistics, similar conclusions can be drawn, that is, the performance superiority of the algorithms keeps consistent with the results in Figs. 4–7. Specifically, the mean and best accuracies of our methods are higher than others in most cases.

5.3. Object recognition

We also prepare an experiment to test the object recognition performance of our SSMM-Isomap method and other related methods. In this study, two benchmark object image databases, i.e., COIL-20 and Caltech, are involved for comparison. The object recognition performance is also compared with those of the linear projection based PCA, LDA, SSLDA, SSMMC, LapLDA and IsoP. The recognition results averaged based on 15 random splits of training and test images are illustrated in Figs. 7 and 8. In this study, two experimental configurations are prepared. The first one is to evaluate the effects of different numbers of training images on the recognition performance; and other one is to fix the number of reduced dimension and evaluate the effects of different numbers of training images on the performance. We have the following observations from the results similarly. (1) The object recognition performance of each method can be improved by the increasing numbers of reduced dimensionalities and training object images. (2) Our SSMM-Isomap1 and SSMM-Isomap2 deliver the comparable and even higher accuracies than their competitors in most cases. The results of SSMM-Isomap1 and SSMM-Isomap2 are close over the COIL-20 database, while SSMM-Isomap2 can consistently outperform SSMM-Isomap1 over the Caltech object database. One can also note that our SSMM-Isomap1 and SSMM-Isomap2 can outperform the others by about 3% improvement on the Caltech object database. The recognition results of LDA, SSLDA and LapLDA are comparative with each other and all are better than other remaining methods in most cases. SSMMC performs poorly based on the two object databases, since its performance is usually worse than others. The result of IsoP is the worst one on the Caltech database.

According to the quantitative evaluation results in Figs. 8 and 9, we summarize the statistics in Tables 9 and 10 respectively, where we similarly show the mean accuracies with standard deviation (std) and the highest ones. Note that the highest two records

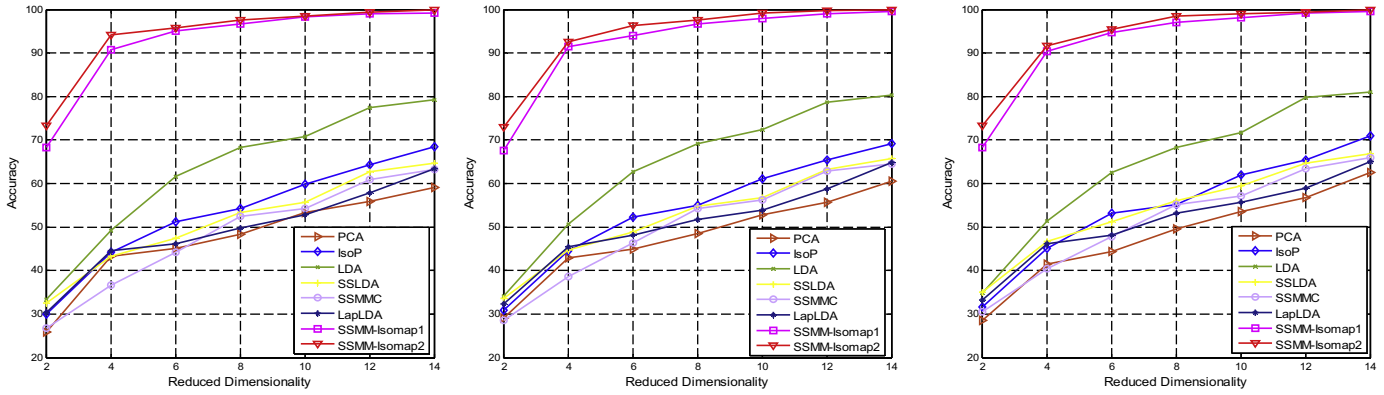


Fig. 4. Classification accuracies vs. varied dimensionalities on YALE face database, where the number of labeled face images is fixed as 3 (*left*), 4 (*middle*) and 5 (*right*), respectively.

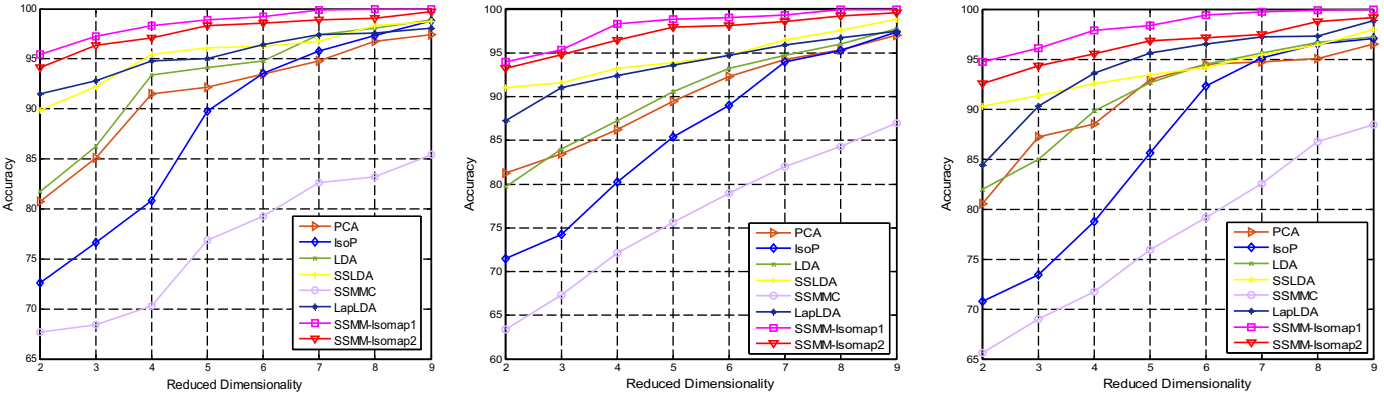


Fig. 5. Classification Accuracies vs. varied dimensionalities on JAFFE face database, where the number of labeled face images is fixed as 6 (*left*), 8 (*middle*) and 10 (*right*), respectively.

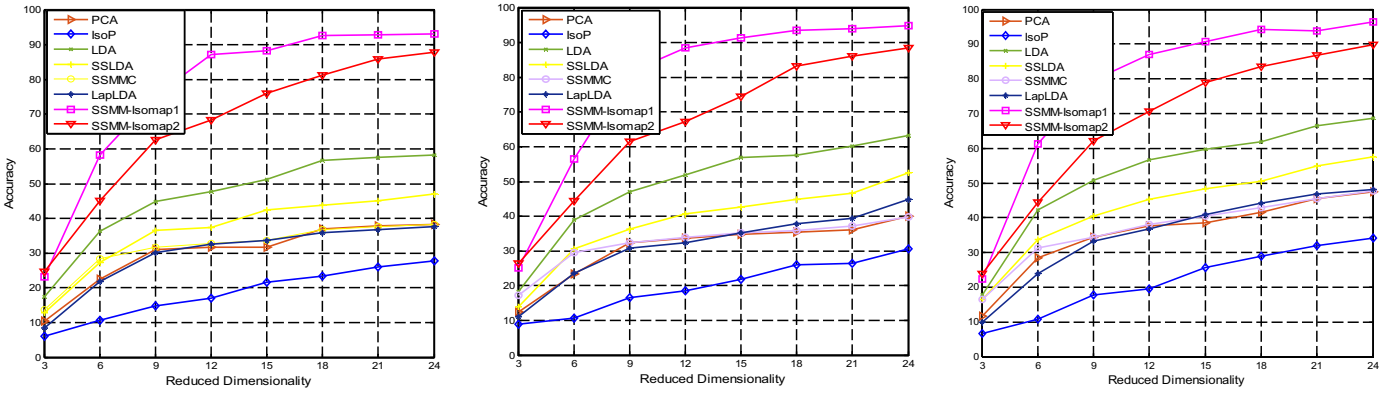


Fig. 6. Classification Accuracies vs. varied dimensionalities on AR-male face database, where the number of labeled face images is fixed as 8 (*left*), 10 (*middle*) and 12 (*right*), respectively.

Table 5
Face recognition comparisons on the YALE face database.

Methods	Settings					
	YALE (3 labeled)		YALE (4 labeled)		YALE (5 labeled)	
	Mean ± *std* (%)	*Best* (%)	*Mean* ± *std* (%)	*Best* (%)	*Mean* ± *std* (%)	*Best* (%)
PCA	47.26 ± 0.1105	59.17	47.80 ± 0.1025	60.49	48.08 ± 0.1121	62.54
IsoP	53.16 ± 0.1307	68.48	54.04 ± 0.1314	69.11	54.76 ± 0.1334	71.00
LDA	62.86 ± 0.1651	79.17	64.02 ± 0.1657	80.23	64.23 ± 0.1646	81.12
SSLDA	51.38 ± 0.1127	64.65	52.51 ± 0.1116	65.78	54.30 ± 0.1057	66.89
SSMMC	48.31 ± 0.1323	63.17	50.19 ± 0.1310	64.44	51.46 ± 0.1270	65.97
LapLDA	49.26 ± 0.1064	63.50	50.75 ± 0.1040	64.89	51.48 ± 0.1028	65.06
SSLM-Isomap1	**92.47 ± 0.1104**	**99.17**	**92.30 ± 0.1127**	**99.48**	**92.45 ± 0.1115**	**99.59**
SSLM-Isomap2	**94.10 ± 0.0938**	**100**	**93.99 ± 0.0969**	**100**	**93.89 ± 0.0952**	**100**

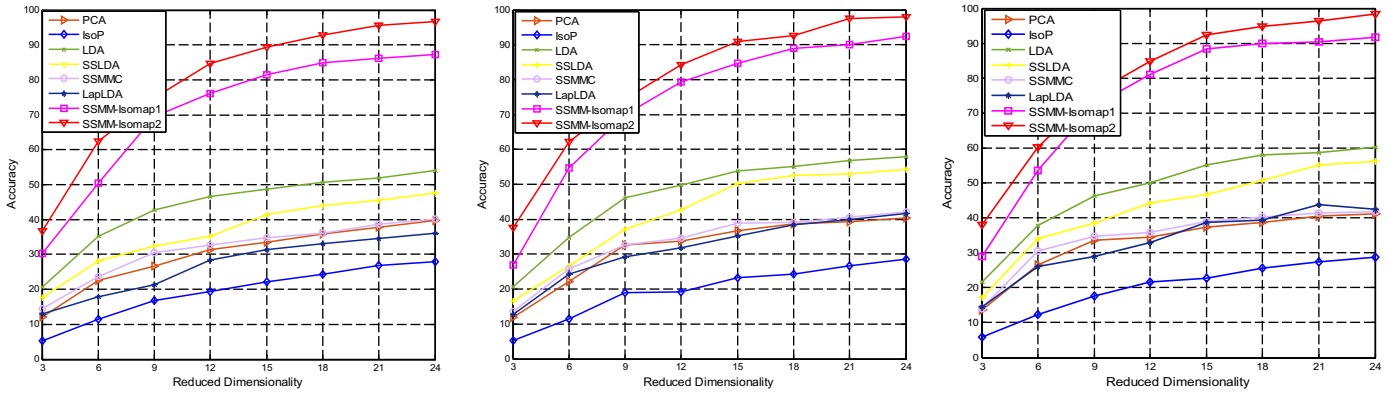


Fig. 7. Classification Accuracies vs. varied dimensionalities on AR-female face database, where the number of labeled face images is fixed as 8 (*left*), 10 (*middle*) and 12 (*right*), respectively.

Table 6
Face recognition comparisons on the JAFFE face database.

Methods	Settings					
	JAFFE (6 labeled)		JAFFE (8 labeled)		JAFFE (10 labeled)	
	Mean ± *std (%)*	*Best (%)*	*Mean* ± *std (%)*	*Best (%)*	*Mean* ± *std (%)*	*Best (%)*
PCA	91.47 ± 0.0578	97.39	89.91 ± 0.0580	96.99	91.23 ± 0.0542	96.49
IsoP	88.16 ± 0.1011	98.77	85.83 ± 0.0978	97.35	89.19 ± 0.1066	97.05
LDA	93.07 ± 0.0605	98.86	90.41 ± 0.2179	97.75	91.66 ± 0.0564	97.23
SSLDA	95.44 ± 0.0301	98.69	94.65 ± 0.028	98.88	93.95 ± 0.0258	97.95
SSMMC	76.70 ± 0.0708	85.43	76.33 ± 0.0831	86.98	77.39 ± 0.0846	88.46
LapLDA	95.44 ± 0.2374	98.04	93.63 ± 0.0339	97.48	94.21 ± 0.0476	98.81
SSMM-Isomap1	**98.62 ± 0.160**	**99.98**	**98.10 ± 0.2224**	**99.98**	**98.24 ± 0.194**	**100**
SSMM-Isomap2	**97.77 ± 0.1836**	**99.75**	**97.24 ± 0.2233**	**99.59**	**96.47 ± 0.2211**	**99.12**

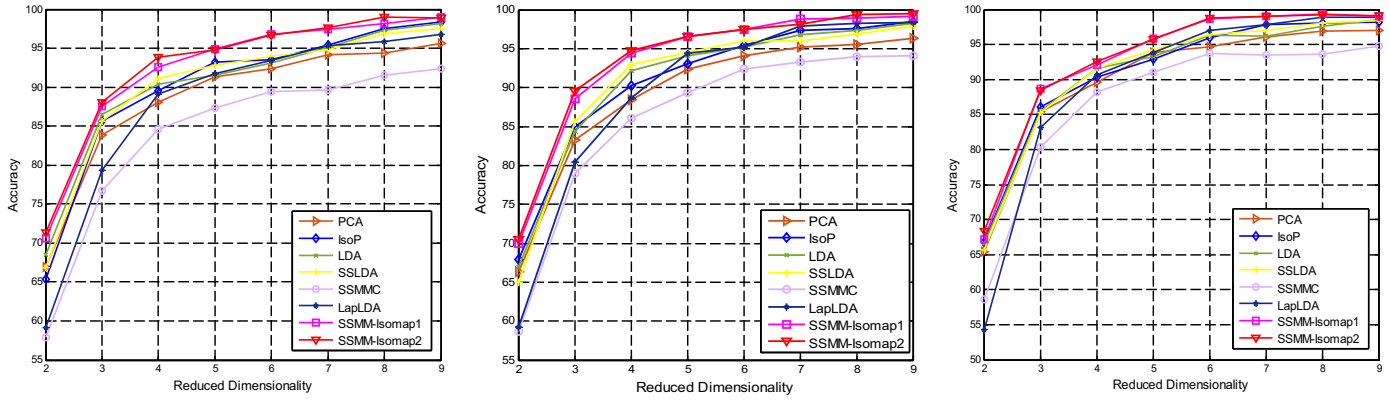
Table 7
Face recognition comparisons on the AR-male face database.

Methods	Settings					
	AR-male (8 labeled)		AR-male (10 labeled)		AR-male (12 labeled)	
	Mean ± *std (%)*	*Best (%)*	*Mean* ± *std (%)*	*Best (%)*	*Mean* ± *std (%)*	*Best (%)*
PCA	30.04 ± 0.0946	38.33	31.04 ± 0.0881	40.13	35.64 ± 0.1140	47.47
IsoP	18.40 ± 0.0767	27.78	19.93 ± 0.0759	30.50	21.95 ± 0.0995	34.14
LDA	46.20 ± 0.1386	58.22	49.24 ± 0.1474	63.25	53.06 ± 0.1662	68.62
SSLDA	36.53 ± 0.1142	4700	38.54 ± 0.1193	52.63	43.45 ± 0.1327	57.43
SSMMC	31.59 ± 0.0797	38.33	32.61 ± 0.0697	39.75	37.10 ± 0.0991	47.62
LapLDA	29.61 ± 0.0992	37.67	31.86 ± 0.1051	44.88	35.48 ± 0.1305	48.14
SSMM-Isomap1	**76.33 ± 0.2456**	**93.11**	**78.14 ± 0.2489**	**94.88**	**78.13 ± 0.2527**	**96.29**
SSMM-Isomap2	**66.42 ± 0.2192**	**87.89**	**66.46 ± 0.2182**	**88.50**	**67.50 ± 0.2308**	**89.86**

Table 8
Face recognition comparisons on the AR-female face database.

Methods	Settings					
	AR_female (8 labeled)		AR_female (10 labeled)		AR_female (12 labeled)	
	Mean ± *std (%)*	*Best (%)*	*Mean* ± *std (%)*	*Best (%)*	*Mean* ± *std (%)*	*Best (%)*
PCA	29.93 ± 0.0921	39.67	31.96 ± 0.0996	40.38	33.26 ± 0.0917	41.14
IsoP	19.26 ± 0.0784	28.00	19.67 ± 0.0791	28.50	20.23 ± 0.0791	28.71
LDA	43.80 ± 0.1112	54.11	46.91 ± 0.1298	58.00	48.46 ± 0.1317	60.29
SSLDA	36.51 ± 0.1020	47.67	41.64 ± 0.1392	54.38	42.79 ± 0.1287	56.14
SSMMC	31.35 ± 0.0855	40.22	33.44 ± 0.0961	42.15	34.67 ± 0.0919	41.86
LapLDA	26.93 ± 0.0856	36.00	31.69 ± 0.0963	41.75	33.32 ± 0.0986	43.71
SSMM-Isomap1	**70.77 ± 0.2036**	**87.22**	**73.35 ± 0.2259**	**92.38**	**74.50 ± 0.2246**	**91.71**
SSMM-Isomap2	**79.14 ± 0.2071**	**96.67**	**79.67 ± 0.2095**	**97.88**	**80.07 ± 0.2126**	**98.29**

under each experimental setting are still highlighted in bold. It can be easily noticed that similar object recognition performance superiority of the algorithms can be found from the results in Tables 9 and 10. That is, our SSMM-Isomap1 and SSMM-Isomap2 can deliver higher mean and better records than the other compared algorithms in most cases. The mean results of existing LDA, SSLDA and LapLDA are comparable with each other under the same experimental setting. IsoP can also be able to obtain the comparable recognition results with LDA, SSLDA and LapLDA over the COIL-20 database.



(a) 20 labeled (b) 25 labeled (c) 30 labeled

Fig. 8. Classification Accuracies vs. varied dimensionalities on COIL-20 object image database, where the number of labeled face images is fixed as 20 (*left*), 25 (*middle*) and 30 (*right*), respectively.

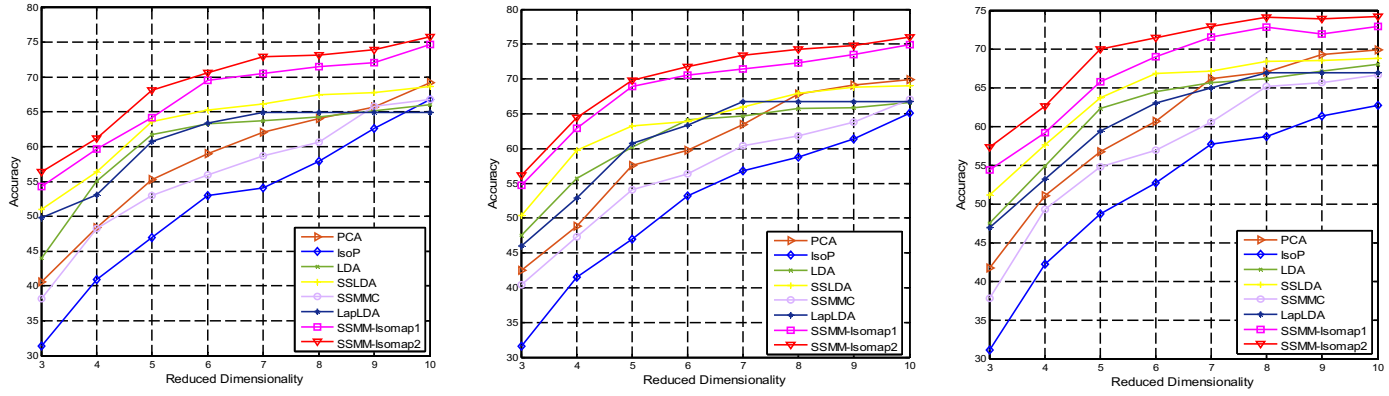


Fig. 9. Classification Accuracies vs. varied dimensionalities on the Caltech image database, where the number of labeled face images is fixed as 80 (*left*), 100 (*middle*) and 120 (*right*), respectively.

Table 9
Objective recognition comparisons on the COIL-20 object image database.

Methods	Settings					
	COIL-20 (20 labeled)		COIL-20 (25 labeled)		COIL-20 (30 labeled)	
	Mean ± *std* (%)	*Best* (%)	*Mean* ± *std* (%)	*Best* (%)	*Mean* ± *std* (%)	*Best* (%)
PCA	88.35 ± 0.0948	95.67	88.96 ± 0.1012	96.38	89.86 ± 0.1064	97.02
IsoP	89.82 ± 0.1072	98.33	90.63 ± 0.1023	98.51	90.75 ± 0.1060	98.17
LDA	90.07 ± 0.0952	98.17	90.64 ± 0.1053	98.23	90.71 ± 0.1059	98.53
SSLDA	89.96 ± 0.1006	97.50	90.62 ± 0.1103	97.87	90.83 ± 0.1106	98.45
SSMMC	83.69 ± 0.1155	92.40	85.84 ± 0.1211	94.09	86.72 ± 0.1230	94.76
LapLDA	87.61 ± 0.1278	96.83	89.05 ± 0.1354	98.41	89.30 ± 0.1514	98.93
SSLM-Isomap1	**92.13 ± 0.0947**	**99.05**	**92.99 ± 0.0996**	**99.17**	**92.49 ± 0.1095**	**99.32**
SSLM-Isomap2	**92.53 ± 0.0927**	**98.98**	**93.21 ± 0.0973**	**99.46**	**92.64 ± 0.1055**	**99.18**

Table 10
Objective recognition comparisons on the Caltech image database.

Methods	Settings					
	Caltech (80 labeled)		Caltech (100 labeled)		Caltech (120 labeled)	
	Mean ± *std* (%)	*Best* (%)	*Mean* ± *std* (%)	*Best* (%)	*Mean* ± *std* (%)	*Best* (%)
PCA	58.04 ± 0.0958	69.20	59.86 ± 0.0994	69.85	60.37 ± 0.0995	69.91
IsoP	51.69 ± 0.1163	66.82	51.91 ± 0.1123	65.15	51.93 ± 0.1085	62.73
LDA	60.40 ± 0.0742	65.97	61.30 ± 0.0665	66.59	62.06 ± 0.0719	68.08
SSLDA	63.28 ± 0.0630	68.65	63.64 ± 0.0625	69.07	64.07 ± 0.0640	68.84
SSMMC	55.89 ± 0.0946	66.75	56.40 ± 0.0892	67.08	57.14 ± 0.0988	66.73
LapLDA	60.86 ± 0.0605	64.95	61.27 ± 0.0783	66.79	61.07 ± 0.0745	66.98
SSMM-Isomap1	**67.06 ± 0.0704**	**74.66**	**68.65 ± 0.0668**	**74.88**	**67.26 ± 0.0697**	**72.98**
SSMM-Isomap2	**69.03 ± 0.0683**	**75.78**	**70.09 ± 0.0671**	**75.98**	**69.61 ± 0.0627**	**74.26**

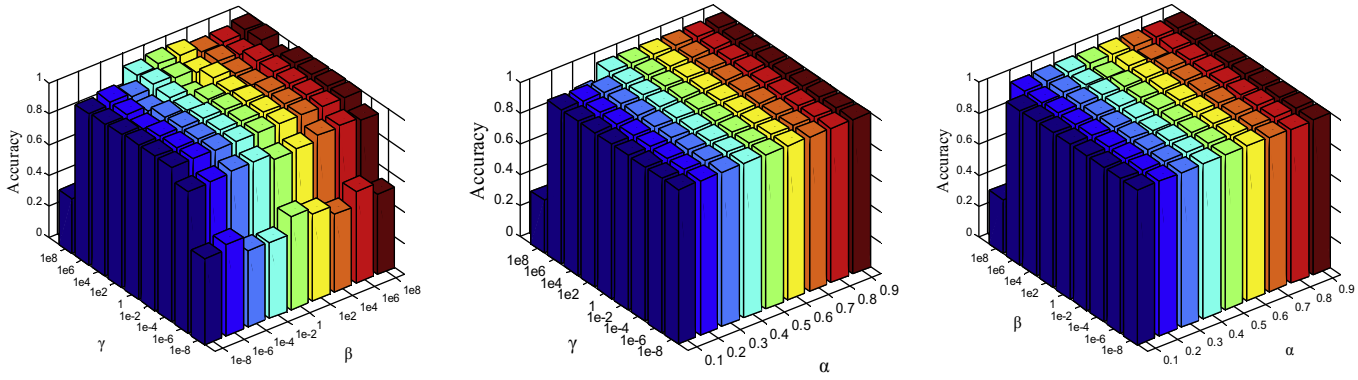


Fig. 10. Classification performance of our SSMM-Isomap under various parameters on the JAFFE face database, where: (*left*) Fix α to tune β andγ; (*middle*) Fix β to tune α andγ; (*right*) Fix γ to tune α andβ.

5.4. Investigation of parameter selections

We investigate the effects of different parameter selections on the performance of our SSMM-Isomap1 using the experimental investigation in this section, since the optimal determination of the model parameters still remains an open problem. In this study, the influence of the model parameters on classification, clustering, and visualization are discussed respectively.

5.4.1. Influence of parameters on the classification results

In this study, JAFFE face database is used for evaluations. We choose 8 labeled samples of each person in the face database to form the training set and test on the rest of the images. Note that there are three model parameters in our objective function, i.e., α, β and γ, but tuning the three parameters at the same time is not easy. As a widely-used method for the parameter selection, the strategy of grid search [31,45–47] is employed. Specifically, we aim at fixing one parameter and tuning the other two by grid search. We explore the effects of different parameter settings by using the following three cases: (1) Fix $\alpha = 0.5$, tune β and γ from candidate set $\{10^{-8}, 10^{-6},\ldots,10^{8}\}$; (2) Fix $\beta = 10^2$, tune α from candidate set {0.1,0.2,...,0.9} and vary γ from $\{10^{-8},10^{-6},\ldots,10^{8}\}$; (3) Fix $\gamma = 10^0$, tune α from candidate set {0.1, 0.2,...,0.9}, and tune β from $\{10^{-8}, 10^{-6}, \ldots, 10^{8}\}$. In each study, we extract features of *c-1* dimension (*c* is the number of classes in the dataset) for our SSMM-Isomap. The parameter selection results on classification are shown in Fig. 10. We can observe that SSMM-Isomap can perform well in a wide range of parameter settings, which is an encouraging message, since the performance of our method is less sensitive to the parameters, i.e., the selection of model parameters will be relatively easier for the real applications. One can note that our SSMM-Isomap are more robust to the model parameters α, β thanγ. Specifically, relatively larger or smaller values of γ may produce decreased performance, as can be found from the results. Note that the above simulation results and analysis can provide a guidance of the selection of model parameters. Note that similar findings about parameter selections can be found from other datasets in most cases, but the experimental results will not be presented because of the page limitation.

5.4.2. Influence of parameters on the clustering results

We then investigate the effects of the model parameters on the clustering results. In this simulation, we randomly choose 3 unlabeled data and 3 labeled data from each class to form the training set and test on the rest. We use the averaged NMI values obtained by performing 30 times *k*-means clustering as the quantitative evaluation metric and the results are shown in the vertical axis of Fig. 11. Similarly, we also investigate the effects of different parameter settings using the same cases used in the above parameter analysis on classification. Results over the three conditions are shown in Fig. 11, from which we can similarly find that the results of our algorithm are also less sensitive to the model parameters. Note that the finding from the parameter analysis results on the clustering keep consistent with that on classification, which is another one encouraging message for the parameter selections. Specifically, our SSMM-Isomap are also more robust to parameters α, β than γ, and moreover relatively larger or smaller values of γ may produce the decreased results.

Table 11
Parameter settings used for visualization.

Case 1	fix$\beta = 10^2$,$\gamma = 10^0$	tune$\alpha = \{0.2, 0.5, 0.8\}$
Case 2	fix$\alpha = 0.5$,$\gamma = 10^0$	tune$\beta = \{10^{-4}, 10^0, 10^4\}$
Case 3	fix$\alpha = 0.5$,$\beta = 10^2$	tune$\gamma = \{10^{-8}, 10^0, 10^8\}$

5.4.3. Influence of parameters on the visualization results

Finally, we investigate the effects of the model parameters on the data visualization results. In this study, COIL-20 object database is employed. Specifically, we choose five categories from the object database, and then randomly choose 8 labeled and 8 unlabeled samples from the 72 images of each class to form the training set. After obtaining the linear projection *P*, the 2D embedding can be obtained by embedding the data onto *P*. Note that it is impossible to visualize all the possible results under different parameter settings. In this study, we mainly discussed the following three cases, i.e., exploring the effects of one parameter on the visualization results by fixing the other two. The used settings as examples are described in Table 11. Note that the parameter analysis results on the visualizations are shown in Fig. 12, where each row presents the result under each one of the evaluated cases.

We can see from the visualization results that: (1) When fixing $\beta = 10^2$ and $\gamma = 10^0$to tune α, the results are less sensitive to different choice of α, since our method with different α can perform well in each setting; (2) When fixing $\alpha = 0.5$ and $\gamma = 10^0$ to tune β, we can see that our method also performs well in most cases, especially for the case of relatively large β; (3) When fixing $\alpha = 0.5$ and $\beta = 10^2$ to tune γ, we ca easily find that our proposed SSMM-Isomap method with $\gamma = 10^0$ tends to outperform the other two settings. It should be noted that generally the above visual analysis results can also keep consistent with the quantitative classification and clustering evaluation results. Therefore, based on the above parameter analysis results on classification, clustering and

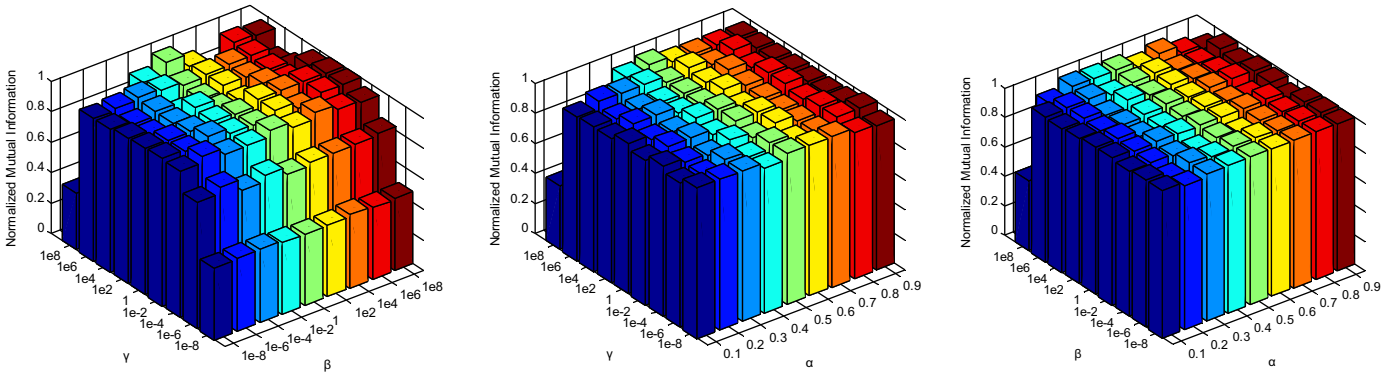


Fig. 11. Clustering performance of our SSMM-Isomap under various parameters on the JAFFE face database, where: (*left*) Fix α to tune β andγ; (*middle*) Fix β to tune α andγ; (*right*) Fix γ to tune α andβ.

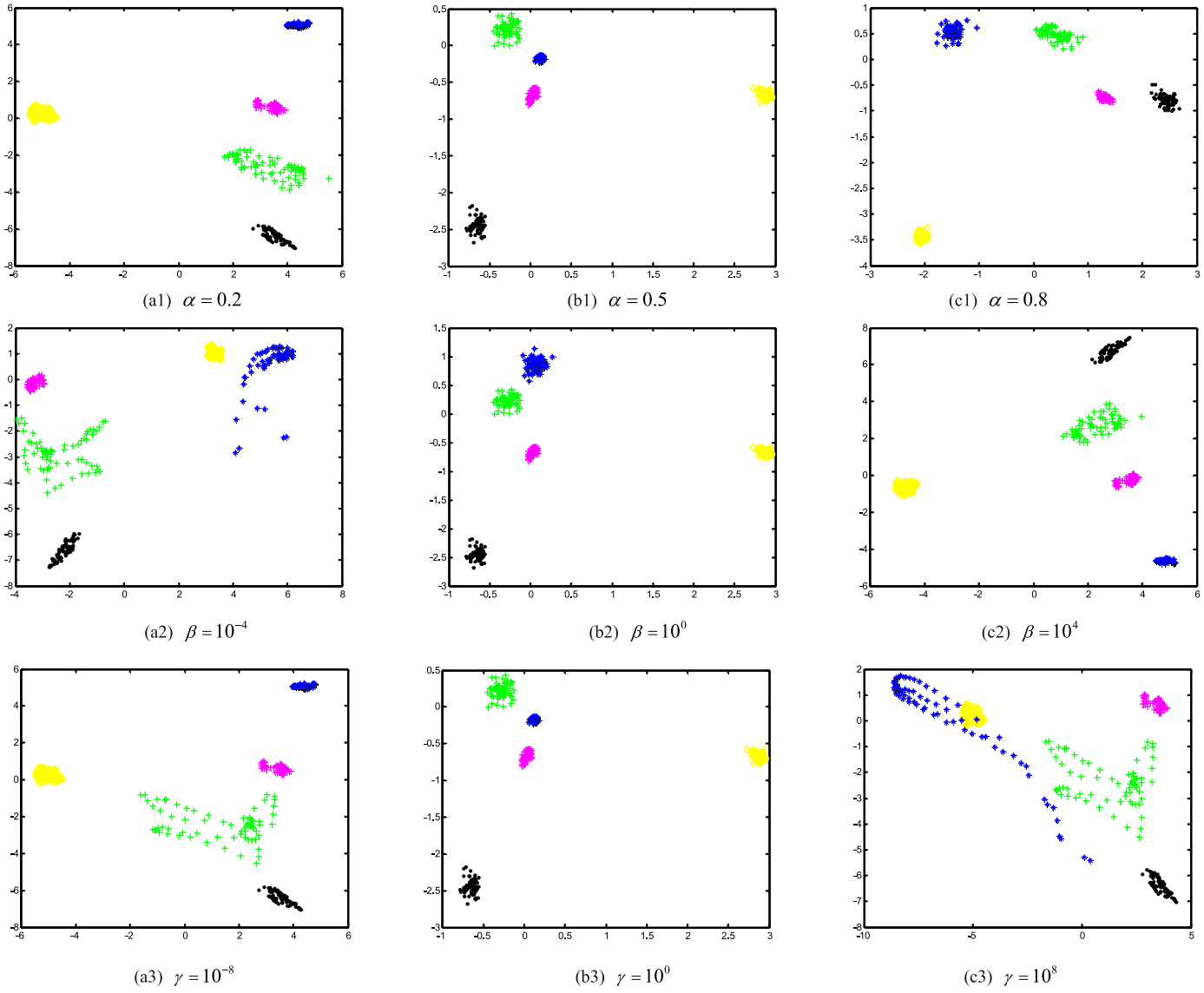


Fig. 12. Visual comparison of the visualization results on COIL-20 database over different parameter selections.

visualization, we can suggest that parameter α can be simply set to 0.5, the range of the parameter γ can be chosen from the region $[10^{-4}, ..., 10^{4}]$, and the parameter β can be chosen from a wide range. Note that we simply set the parameters α, β, γ to 0.5, 10^{2} and 1 respectively for our simulations.

6. Concluding Remarks

In this paper, we have discussed the semi-supervised manifold feature learning problem and moreover have presented a semi-supervised local multi-manifold Isomap by linear embedding technically. The proposed formulation can use both labeled and

unlabeled data for multi-manifold feature learning. More specifically, our proposed formulation aims at minimizing the pairwise intra-class distances on the same manifold and maximizing the distances between data points on different manifolds. Besides, a neighborhood reconstruction error term is incorporated to preserve local neighborhood information of training samples. To make our model applicable to outside new data, a feature approximation error that can correlate local multi-manifold features with the embedded features by a linear projection is also included into the nonlinear manifold learning framework, so the out-of-sample extension issue can be naturally solved.

Extensive clustering and classification results demonstrate the validity of our algorithms for image feature extraction and recognition. In future, investigating the joint formulation of manifold feature leaning and reconstruction weights learning will be considered, because the pre-calculated reconstruction weights cannot be ensured to be optimal for the subsequent feature leaning. Besides, the optimal determination of model parameters still remains an open problem in reality, which needs further investigation in future.

Acknowledgements

This work is partially supported by National Natural Science Foundation of China (61402310, 61672365 and 61672364), Major Program of Natural Science Foundation of Jiangsu Higher Education Institutions of China (15KJA520002), Special Funding of China Postdoctoral Science Foundation (2016T90494), Postdoctoral Science Foundation of China (2015M580462), Postdoctoral Science Foundation of Jiangsu Province of China (1501091B), and the High-Level Talent of the "Six Talent Peak Project" of Jiangsu Province of China (XYDXX-055).

Appendix A. The procedure of majorization

The main idea of SMACOF is to minimize a more simple and manageable auxiliary function $g(x,z)$instead of minimizing the complicated function $f(x)$. These two functions $f(x)$and $g(x,z)$should satisfy the following three conditions: (1) $f(x) \le g(x,z)$for all x and z; (2) $f(x) = g(x,x)$for each x; (3) $g(x,z)$should be simpler to be minimized than $f(x)$. Note that the function $f(x)$has been proved to be monotonically non- increasing [11]. The detailed procedures of majorization has been described in the Table 12.

Table 12
The procedure of majorization.

Inputs: Initial value x_0 and a small p°sitive c°nstant ξ.
Step 1: Compute the function value $f(x_0)$, and set step $n=0$;
Step 2: Substituting $z = x_n$ into $g(x,z)$ to obtain $g(x,x_n)$;
Step 3: Minimize $g(x,x_n)$ *w.r.t.* x, and assume the minimum is obtained at x^*, then set $x_{n+1} = x^*$ to gain the function value $f(x_{n+1})$;
Step 4: If $f(x_n) - f(x_{n+1}) < \xi$, stop, else set $n = n+1$, go to **Step 2**.
Outputs: The solution x_n.

Appendix B. Detailed procedures of using SMACOF to solve our SSMM-Isomap

We mainly show the detailed procedures of using SMACOF to solve our optimization problem in Eq. (10). To do this, we solve Eqs. (13)–(16) individually. To solve Eq. (13), let

$$w_{ij}^{ML} = \begin{cases} \frac{1-\alpha}{|ML|} & if\left(x_i, x_j\right) \in ML \\ 0 & otherwise \end{cases}, w_{ij}^{CL} = \begin{cases} \frac{\alpha}{|CL|} & if\left(x_i, x_j\right) \in CL \\ 0 & otherwise \end{cases}, \tag{33}$$

$$V_{ML} = \sum_{i,j=1, i\neq j}^{N} w_{ij}^{ML} A^{ij},\ V_{CL} = \sum_{i,j=1, i\neq j}^{N} w_{ij}^{CL} A^{ij},\ V = V_{ML} - V_{CL}, \tag{34}$$

then we can rewrite Eq. (13) as follows based on *Remark 1* in [9]:

$$\begin{aligned} \eta^2(Y) &= \sum_{i,j=1, i\neq j}^{N} w_{ij}^{ML} tr\left(YA^{ij}Y^T\right) - \sum_{i,j=1, i\neq j}^{N} w_{ij}^{CL} tr\left(YA^{ij}Y^T\right) \\ &= tr\left(Y\left(\sum_{i,j=1, i\neq j}^{N} w_{ij}^{ML} A^{ij}\right)Y^T\right) - tr\left(Y\left(\sum_{i,j=1, i\neq j}^{N} w_{ij}^{CL} A^{ij}\right)Y^T\right) \\ &= tr\left(YV_{ML}Y^T\right) - tr\left(YV_{CL}Y^T\right) \\ &= tr\left(YVY^T\right). \end{aligned} \tag{35}$$

Next, we show the optimization of Eq. (14). Since $1-\alpha > 0$ and $d_G(x_i, x_j) > 0$, based on the *Remark 2* in [9], we can have

$$\begin{aligned} \rho(Y) &= \frac{1-\alpha}{|ML|} \sum_{(x_i,x_j)\in ML} d\left(y_i, y_j\right) d_G\left(x_i, x_j\right) \\ &\ge \frac{1-\alpha}{|ML|} \sum_{\substack{(x_i,x_j)\in ML \\ d\left(z_i,z_j\right)\neq 0}} d\left(y_i, y_j\right) d_G\left(x_i, x_j\right) \\ &\ge \sum_{\substack{(x_i,x_j)\in ML \\ d\left(z_i,z_j\right)\neq 0}} \frac{(1-\alpha) d_G\left(x_i, x_j\right) tr\left(YA^{ij}Z^T\right)}{|ML| d\left(z_i, z_j\right)} \\ &\ge tr\left(Y\left(\sum_{i,j=1, i\neq j}^{N} w_{ij} A^{ij}\right)Z^T\right). \end{aligned} \tag{36}$$

Suppose that the weights w_{ij} satisfy the following definition for each sample pair:

$$w_{ij} = \begin{cases} \frac{(1-\alpha) d_G\left(x_i, x_j\right)}{|ML| d\left(z_i, z_j\right)} & if\left(x_i, x_j\right) \in ML \text{ and } d\left(z_i, z_j\right) \neq 0 \\ 0 & otherwise \end{cases}. \tag{37}$$

According to the result in the *Remark 3* in [9], i.e., $\sum_{i,j=1, i\neq j}^{N} w_{ij} A^{ij} = B$, which is clearly dependent on Z, so we can rewrite B as $B(Z)$ and rewrite the formulation in Eq. (37) as

$$\rho(Y) \ge tr\left(Y\left(\sum_{i,j=1, i\neq j}^{N} w_{ij} A^{ij}\right)Z^T\right) = tr\left(YB(Z)Z^T\right). \tag{38}$$

Then, we can transform Eq. (15) into the following form:

$$\begin{aligned} \omega(Y) &= \beta \left\| Y - YW^T \right\|_F^2 = \beta tr\left(\left(Y - YW^T\right)\left(Y - YW^T\right)^T\right) \\ &= \beta tr\left(Y\left(I - W^T\right)(I - W)Y^T\right), \end{aligned} \tag{39}$$

where W can be obtained by Eq. (9), I is an identity matrix of size $N \times N$. Let $M = \beta(I - W^T)(I - W)$, then we can reformulate the equation in Eq. (39) as

$$\omega(Y) = tr\left(YMY^T\right). \tag{40}$$

Finally, we show how to solve the problem of Eq. (16). The projection P in Eq. (16) can be easily obtained by Eq. (17). By substituting Eq. (17) into Eq. (16), we can easily obtain

$$\begin{aligned} \varphi(Y, P) &= \gamma \left\| PX - Y \right\|_F^2 = \gamma tr\left((PX - Y)(PX - Y)^T\right) \\ &= \gamma tr\left(-YX^T\left(XX^T\right)^{-1}XY^T + Y^TY\right) \\ &= \gamma tr\left(Y\left(I - X^T\left(XX^T\right)^{-1}X\right)Y^T\right). \end{aligned} \tag{41}$$

Let $K=\gamma(I-X^T(XX^T)^{-1}X)$, then the formulation in Eq. (41) can be rewritten in more concise form as

$$\varphi(Y,P)=tr\left(YKY^T\right). \tag{42}$$

By combining the equations in Eqs. (12), (35), (39), (40) and (42), our criterion in Eq. (10) can be transformed into

$$\begin{aligned} J &= \min_{Y,P}\sigma(Y,P)\\ \sigma(Y,P) &= \eta^2{}_{con}+\eta^2(Y)-2\rho(Y)+\omega(Y)+\varphi(Y,P)\\ &\le \eta^2{}_{con}+tr\left(YVY^T\right)-2tr\left(YB(Z)Z^T\right)+tr\left(YMY^T\right)\\ &\quad +tr\left(YKY^T\right)\\ &= \eta^2{}_{con}+tr\left(Y(V+M+K)Y^T\right)-2tr\left(YB(Z)Z^T\right)\\ &= \tau(Y,Z), \end{aligned} \tag{43}$$

where $\tau(Y,Z)=\eta^2{}_{con}+tr(Y(V+M+K)Y^T)-2tr(YB(Z)Z^T)$ denotes an auxiliary function. Then, low-dimensional features Y can be obtained by setting the derivative of Eq. (18) *w.r.t.* Y to 0:

$$\begin{aligned} &\partial\tau(Y,Z)/\partial Y=2Y(V+M+K)-2ZB(Z)=0\\ &\Rightarrow Y=ZB(Z)(V+M+K)^{-1}. \end{aligned} \tag{44}$$

References

[1] S. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326.
[2] I.T. Jolliffe, in: Principal Component Analysis, 87, Springer, Berlin, 1986, pp. 41–64.
[3] J.B. Tenenbaum, V. de Silva, J.C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (5500) (2000) 2319–2323.
[4] T.F. Cox, M.A Cox, Multidimensional Scaling, CRC Press, 2000.
[5] D.D. Ridder, O. Kouropteva, O. Okun, M. Pietikanien, R.W. Duin, Supervised locally linear embedding, in: Proceedings of the Artificial Neural Networks and Neural Information Processing, 2003.
[6] X. Geng, D.C. Zhan, Z.H. Zhou, Supervised nonlinear dimensionality reduction for visualization and classification, IEEE Trans. Syst. Man Cybern. Part B 35 (6) (2005) 1098–1107.
[7] W.K. Wonga, H.T. Zhao, Supervised optimal locality preserving projection, Pattern Recognit. 45 (1) (2012) 186–197.
[8] Z. Zhang, T. Chow, M. Zhao, M-Isomap: Orthogonal constrained marginal isomap for nonlinear dimensionality reduction, IEEE Trans. Cybern. 43 (1) (2013) 180–191.
[9] B. Yang, M. Xiang, Y. Zhang, Multi-manifold Discriminant Isomap for visualization and classification, Pattern Recognit. 55 (2016) 215–230.
[10] Z. Zhang, M. Zhao, T.W.S. Chow, Marginal semi-supervised sub-manifold projections with informative constraints for dimensionality reduction and recognition, Neural Netw. 36 (2012) 97–111.
[11] M. Belkin, P. Niyogi, Semi-supervised learning on Riemannian manifolds, Mach. Learn. 56 (1-3) (2004) 209–239.
[12] X. Zhu, Semi-supervised Learning Literature Survey, University of Wisconsin Madison, 2008 Computer Sciences TR 1530.
[13] Y. Song, F. Nie, C. Zhang, S. Xiang, A unified framework for semi-supervised dimensionality reduction, Pattern Recognit. 41 (9) (2008) 2789–2799.
[14] L. Zhang, L. Wang, W. Lin, Semisupervised biased maximum margin analysis for interactive image retrieval, IEEE Trans. Image Process. 21 (4) (2012) 2294–2308.
[15] O. Chapelle, B. Scholkopf, A. Zien, Semi-supervised learning, IEEE Trans. Neural Netw. 20 (30) (2009) 542-542.
[16] M. Belkin, P. Niyogi, Laplacian eigenmaps and spectral techniques for embedding and clustering, Neural Inf. Process. Syst. 14 (6) (2002) 585–591.
[17] G. Rosman, M.M. Bronstein, A.M. Bronstein, R. Kimmel, Nonlinear dimensionality reduction by topologically constrained isometric embedding, Int. J. Comput. Vis. 89 (1) (2010) 56–68.
[18] T. Qiao, Y. Zhang, H. Liu, Nonlinear expectation maximiza- tion estimator for TDOA localization, IEEE Wirel. Commun. Lett. 3 (6) (2014) 637–640.
[19] J. de Leeuw, P. Mair, Multidimensional scaling using majorization: SMACOF in R, J. Stat. Softw. 31 (3) (2009) 1–30.
[20] C. Moeglin, J.L. Waldspurger, Spectral Decomposition and Eisenstein Series: A Paraphrase of the Scriptures, Cambridge University Press, 1995.
[21] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Comput. 15 (6) (2003) 1373–1396.
[22] Y. Zhang, Z. Zhang, L. Jia, F. Li, Joint nuclear-norm nonlinear manifold learning and robust classification by linear embedding, in: Proceedings of the IEEE International Joint Conference on Neural Networks, Vancouver, Canada,, 2016 July.
[23] M. Sniedovich, Dynamic Programming: Foundations and Principles, CRC press, 2010.
[24] Q. Wang, M. Chen, X. Li, Quantifying and detecting collective motion by manifold learning, in: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2017, pp. 4292–4298.
[25] S. Hougardy, The Floyd–Warshall algorithm on graphs with negative cycles, Inf. Process. Lett. 110 (8) (2010) 279–281.
[26] S.C. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang, S. Lin, Graph embedding and extensions: a general framework for dimensionality reduction, IEEE Trans. Pattern Anal. Mach. Intell. 29 (1) (2007) 40–51.
[27] D. Sun, D. Zhang, Bagging constraint score for feature selection with pairwise constraints, Pattern Recognit. 43 (6) (2010) 2106–2118.
[28] M.S. Baghshah, S.B. Shouraki, Semi-supervised metric learning using pairwise constraints, in: Proceedings of the International Joint Conference on Artificial Intelligence, 9, 2009, pp. 1217–1222.
[29] F. Wang, Semisupervised metric learning by maximizing constraint margin, IEEE Trans. Syst. Man Cybern. Part B 41 (4) (2011) 931–939.
[30] T. Sim, T. Kanade, Combining models and exemplars for face recognition: An illuminating example, in: *Proceedings of the CVPR 2001 Workshop on Models* versus *Exemplars in Computer Vision*, vol. 1, 2001.
[31] Z. Zhang, F. Li, M. Zhao, Joint low-rank and sparse principal feature coding for enhanced robust representation and visual classification, IEEE Trans. Image Process. 25 (6) (2016) 2429–2443.
[32] S.A. Nene, S.K. Nayar, H. Murase, Columbia Object Image Library (COIL-20), 1996 Technical report CUCS-005-96.
[33] J. Chen, J. Ye, Q. Li, Integrating global and local structures: a least squares framework for dimensionality reduction, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.
[34] D. Cai, X. He, J. Han, Isometric projection, in: Proceedings of the National Conference on Artificial Intelligence, 22, Menlo Park, CA; Cambridge, MA; London, AAAI Press; MIT Press, 2007, p. 528.
[35] D. Cai, X. He, J. Han, Document clustering using locality preserving indexing, IEEE Trans. Knowl. Data Eng. 17 (12) (2005) 1624–1637.
[36] X.F. He, D. Cai, P. Niyogi, Laplacian score for feature selection, Adv. Neural Inf. Process. Syst. (2005) 507–514.
[37] I. Borg, P.J.F. Groenen, Modern Multidimensional scaling: Theory and Applications, Springer Science & Business Media, 2005.
[38] X. He, P. Niyogi, Locality preserving projections, in: Proceedings of Neural Information Processing Systems (NIPS), vol. 16, 2003.
[39] A. Martinez, A.C. Kak, PCA versus LDA, IEEE Trans. Pattern Anal Mach. Intell. 23 (2) (2001) 228–233.
[40] X. He, D. Cai, S. Yan, Neighborhood preserving embedding, in: Proceedings of the IEEE International Conference on Computer Vision, 2005, pp. 1208–1213.
[41] Z. Zhang, T. Chow, M. Zhao, Trace ratio optimization- based semi-supervised nonlinear dimensionality reduction for marginal manifold visualization, IEEE Trans. Knowl. Data Eng. 25 (5) (2013) 1148–1161.
[42] M. Zhao, Z. Zhang, T.W.S. Chow, Trace ratio criterion based generalized discriminative learning for semi-supervised dimensionality reduction, Pattern Recognit. 45 (4) (2012) 1482–1499.
[43] A.M. Martinez, R. Benavente, The AR Face Database, Computer Vision Centre (CVC), Universitat Autònoma de Barcelona, Spain, 1998 Technical Report #24.
[44] J. Schmidhuber, Deep learning in neural networks: an overview, Neural Netw. 61 (2015) 85–117.
[45] W. Jiang, Z. Zhang, F. Li, Joint label consistent dictionary learning and adaptive label prediction for semisupervised machine fault classification, IEEE Trans. Ind. Inf. 12 (1) (2016) 248–256.
[46] M. Muja, D G. Lowe, Scalable nearest neighbor algorithms for high dimensional data, IEEE Trans. Pattern Anal. Mach. Intell. 36 (11) (2014) 2227–2240.
[47] J. Bergstra, D. Yamins, D. Cox, Making a science of model search: hyperparameter optimization in hundreds of dimensions for vision architectures, in: Proceedings of the International Conf. on Machine Learning (ICML), vol. 28, 2013, pp. 115–123.
[48] X. Han, L. Clemmensen, Regularized generalized eigen-decomposition with applications to sparse supervised feature extraction and sparse discriminant analysis, Pattern Recognit. 49 (2016) 43–54.
[49] Z. Zhao, L. Jiao, F. Liu, Semisupervised discriminant feature learning for SAR image category via sparse ensemble, IEEE Trans. Geosci. Remote Sensing 54 (6) (2016) 1–16.
[50] C L. Lekamalage, Y. Yang, G Huang, Dimension reduction with extreme learning machine, IEEE Trans. Image Process. 25 (8) (2016) 1-1.
[51] M. Wang, W. Fu, S. Hao, H. Liu, X. Wu, Learning on big graph: label inference and regularization with anchor hierarchy, IEEE Trans. Knowl. Data Eng. 29 (5) (2017) 1101–1114.
[52] M. Wang, W. Fu, S. Hao, D. Tao, X. Wu, Scalable semi-supervised learning by efficient anchor graph regularization, IEEE Trans. Knowl. Data Eng. 28 (7) (2016) 1864–1877.
[53] M. Wang, X. Liu, X. Wu, Visual classification by l_1-hypergraph modeling, IEEE Trans. Knowl. Data Eng. 27 (9) (2015) 2564–2574.
[54] X. Li, M. Chen, F. Nie, Q. Wang, A multiview-based parameter free framework for group detection, in: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), 2017, pp. 4147–4153.

Yan Zhang is currently working toward the research degree at School of Computer Science and Technology, Soochow University, P. R. China. Her current research interests mainly include pattern recognition, machine learning and data mining.

Zhao Zhang received the Ph.D. degree from the Department of Electronic Engineering (EE), City University of Hong Kong, in 2013. He is currently an Associate Professor at the School of Computer Science and Technology, Soochow University, Suzhou, P. R. China. Dr. Zhang was a Visiting Research Engineer at the National University of Singapore, from Feb to May 2012. He then visited the National Laboratory of Pattern Recognition (NLPR) at Chinese Academy of Sciences (CAS), from Sep to Dec 2012. His current research interests include data mining & statistical machine learning, pattern recognition & image analysis. Dr. Zhang has authored/ co-authored over 60 technical papers published at prestigious international journals and conferences, including IEEE TKDE (3), IEEE TIP (3), IEEE TNNLS (2), ACM TIST, IEEE TSP, IEEE TCYB, IEEE TII (2), Pattern Recognition (4), Neural Networks (5), ICDM, ACM ICMR, ICIP and ICPR, etc. Specifically, he has got 11 first-author regular papers published by the IEEE/ACM Transactions journals. Dr. Zhang is now serving on the Editorial Board of Neurocomputing, IET Image Processing, Neural Processing Letters (NPL), and Neural Computing and Applications (NCA). Besides, he served as a Senior Program Committee (SPC) member of PAKDD 2017, an Area Chair (AC) of BMVC 2016/2015, a Program Committee (PC) member for several popular international conferences (including SDM 2017/2016/2015, PAKDD 2016, IJCNN 2017/2015, CAIP 2015, etc), and also often got invited as a journal reviewer for IEEE TNNLS,IEEE TIP,IEEE TKDE,IEEE TSP,IEEE TCYB, IEEE TMM, IEEE TII,Pattern Recognition,Information Sciences, etc.

Jie Qin received his Bachelor of Science and Ph.D. degrees from the School of Computer Science and Engineering, Beihang University (BUAA), Beijing, P.R. China, in July 2011 and March 2017, respectively. Dr. Qin is currently a Postdoctoral Researcher, under the supervision of Prof. Luc Van Gool, with the Computer Vision Laboratory, ETH Zurich, Switzerland. He was also a visiting researcher at the Department of Electronic and Electrical Engineering, The University of Sheffield, U.K, supervised by Prof. Ling Shao. His current research interests include Computer Vision, Machine Learning, Pattern Recognition and Image/Video Processing. In particular, he is interested in Human Activity Analysis and Content-Based Image/Video Retrieval.

Li Zhang is currently a Full Professor at the School of Computer Science and Technology, Soochow University, Suzhou, China. She obtained her Bachelor and PhD degrees from Xidian University in 1997 and 2002, respectively. She was also a postdoc researcher at the Shanghai Jiaotong University from 2003 to 2005. His research interests include pattern recognition, machine learning, and data mining. Dr. Zhang has authored/ co-authored more than 90 technical papers published at prestigious international journals and conferences, including IEEE TNNLS, IEEE TSP, IEEE Trans. SMC Part B, Information Sciences, Pattern Recognition, Neural Networks, etc.

Bing Li received the Ph.D. degree in computer engineering from the Department of Electronic Engineering at City University of Hong Kong, in 2013. Dr. Li is now an Associate Professor in the School of Economics, Wuhan University of Technology, P. R. China. Her current research interests include machine learning, data mining, and pattern recognition.

Fanzhang Li is now a full Professor and Dean of the School of Computer Science and Technology, Soochow University, P. R. China, and is also an adjunct professor at Beijing Jiaotong University, China. Prof. Li received his Master degree in engineering from the Department of Computer Science, The University of Science and Technology of China, in 1999. His current interests include Lie Group Machine learning, Data Mining, and Dynamic Fuzzy Logic. Prof. Li has published one Monograph Book on Dynamic Fuzzy Logic and its Applications, and has also published more than 100 technical papers in international journals and conferences.