A General Matrix Function Dimensionality Reduction Framework and Extension for Manifold Learning

Ruisheng Ran, Ji Feng, Shougui Zhang, and Bin Fang, *Senior Member, IEEE*

***Abstract*—Many dimensionality reduction methods in the manifold learning field have the so-called small-sample-size (SSS) problem. Starting from solving the SSS problem, we first summarize the existing dimensionality reduction methods and construct a unified criterion function of these methods. Then, combining the unified criterion with the matrix function, we propose a general matrix function dimensionality reduction framework. This framework is configurable, that is, one can select suitable functions to construct such a matrix transformation framework, and then a series of new dimensionality reduction methods can be derived from this framework. In this article, we discuss how to choose suitable functions from two aspects: 1) solving the SSS problem and 2) improving pattern classification ability. As an extension, with the inverse hyperbolic tangent function and linear function, we propose a new matrix function dimensionality reduction framework. Compared with the existing methods to solve the SSS problem, these new methods can obtain better pattern classification ability and have less computational complexity. The experimental results on handwritten digit, letters databases, and two face databases show the superiority of the new methods.**

***Index Terms*—Dimensionality reduction, manifold learning, matrix function, small-sample-size (SSS) problem.**

Manuscript received September 10, 2019; revised January 21, 2020 and April 3, 2020; accepted June 4, 2020. Date of publication July 22, 2020; date of current version April 5, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61876026; in part by the Humanities and Social Sciences Project of the Ministry of Education of China under Grant 20YJAZH084 and Grant 18XJC880002; in part by the Project of Natural Science Foundation Project of CQ CSTC of China under Grant cstc2016jcyjA0419 and Grant cstc2017jcyjAX0316; in part by the School Fund Project of CQNU under Grant 16XLB006 and Grant 16XZH07; and in part by Chongqing Technology Innovation and Application Development Key Project under Grant cstc2019jscx-mbdxX0061. This article was recommended by Associate Editor D. Tao. *(Corresponding author: Bin Fang.)*

Ruisheng Ran is with the College of Computer and Information Science, College of Intelligent Science, Chongqing Normal University, Chongqing 401331, China, and also with the Chongqing Engineering Research Center of Educational Big Data Intelligent Perception and Application, Chongqing Normal University, Chongqing 401331, China (e-mail: rshran@cqnu.edu.cn).

Ji Feng is with the College of Computer and Information Science, Chongqing Normal University, Chongqing 401331, China (e-mail: fengji@cqnu.edu.cn).

Shougui Zhang is with the School of Mathematical Science, Chongqing Normal University, Chongqing 401331, China (e-mail: shgzhang@cqnu.edu.cn).

Bin Fang is with the College of Computer Science, Chongqing University, Chongqing 400044, China (e-mail: fb@cqu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCYB.2020.3003620.

Digital Object Identifier 10.1109/TCYB.2020.3003620

I. Introduction

In THE past few years, dimensionality reduction has been a hot topic in machine learning and computer vision fields. There are many classical dimensionality reduction methods, for example, principal component analysis (PCA) [1]; linear discriminant analysis (LDA) [2], [3]; Laplacian eigenmaps (LE) [4], [5]; ISOMAP [6]; local linear embedding (LLE) [7]; etc. Recently, some new dimensionality reduction methods for clustering or classification have also been proposed [8], [9].

Dimensionality reduction methods can be roughly divided into linear and nonlinear dimensionality reduction [10]. The representative ones of the linear methods are PCA and LDA. The premise of the linear method is that the data are globally linear or linearly separable. But in real life, the actual data are highly distorted and appear more as a nonlinear structure, that is, in a manifold distribution. The classical linear methods cannot work well in this situation, so the nonlinear dimensionality reduction method based on manifold learning came into being. The well-known nonlinear methods are LE [4], [5], ISOMAP [6], LLE [7], local tangent space alignment (LTSA) [11], etc. These manifold learning methods make a mapping from the original dataset to the low-dimensional dataset, and then reveal and visualize the geometry between the original data in a low-dimensional space.

Manifold learning has been widely used in many applications. For example, the t-SNE method is used to reduce dimensionality and visualize the features of high-dimensional data [12]. In [13]–[15], manifold regularization terms, which encode the local geometrical information existing in data, are integrated into the low-rank matrix factorization to construct the objective function for hyperspectral image unsupervised classification, joint clustering and feature extraction, and multiview feature embedding. In [16] and [17], with the graph-based Laplacian, new objection functions are designed and used to image ranking and image retrieval, respectively. In [18], a weighted triplet loss function is proposed for place recognition. This function shares a similar form with the objective of manifold learning. Some new manifold learning methods are also developed. For example, Grassmann hashing for approximate nearest-neighbor search on the Grassmann manifold [19], subspace indexing model on the Grassmann manifold for image search [20], and Grassmannian regularized structured multiview embedding for image classification [21].

However, these classical nonlinear dimensionality reduction methods are all nonlinear mappings; they just make the mapping of the training samples not applicable to the classification problem (e.g., face recognition). In this way, many linearization methods are developed. There are two research routes: 1) unsupervised and 2) supervised. In the unsupervised case, the representative ones are locality preserving projections (LPP) [22]; ISOMAP projections (IsoP) [23]; neighborhood preserving embedding (NPE) [24]; neighborhood preserving projections (NPP) [25], [26]; linear LTSA (LLTSA) [27]; etc. In the supervised case, the representative ones are local discriminant embedding (LDE) [28]; discriminant LPP (DLPP) [29]; locally discriminating projection (LDP) [30]; neighborhood preserving discriminant embedding (NPDE) [31]; marginal Fisher analysis (MFA) [32], [33]; discriminant Isomap projections [34], [35]; etc.

Unfortunately, the above methods, regardless of supervised or unsupervised learning, all have the small-sample-size (SSS) problem. These methods are usually reduced to the solution of the following generalized eigenproblem:

$$\boldsymbol{S}_1\boldsymbol{u} = \lambda \boldsymbol{S}_2\boldsymbol{u} \tag{1}$$

where the two matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ have different forms for different methods. However, in real life, the number of samples is much smaller than the dimension of the sample, resulting in that the matrix $\boldsymbol{S}_2$ is singular and then (1) is unsolvable. This is a so-called SSS problem.

A common way to solve the SSS problem is to reduce the dimensions of the original sample using the PCA method, and then the LPP, NPE, DLPP, LDE, NPDE, and MFA methods are executed in such a reduced sample space. But due to the dimensionality reduction at the PCA stage, some important information is discarded.

Another way to solve the SSS problem is to apply the idea of regularization, by adding some constant values to the diagonal elements of the matrix $\boldsymbol{S}_2$ in (1), as $\alpha\boldsymbol{I} + \boldsymbol{S}_2$, where $\alpha > 0$. Then, the matrix $\alpha\boldsymbol{I} + \boldsymbol{S}_2$ is nonsingular, so (1) is solvable. This method has been proposed in [36] for the LPP method. More generally, this method is given in [37] and is universally applied to LPP, NPE, and IsoP methods. For DLPP, a regularized generalized DLPP method (RGDLPP), which may be viewed as a variant of the regularization method, has been presented to solve the SSS problem of DLPP [38]. Besides, the method based on the maximum margin criterion (MMC) is a simple and common way to solve the SSS problem. It may also be referred to in [36]. In [36], this method is called "LPPImprove3," but in fact it comes from the idea of MMC. With MMC, the maximum margin NPE method (MMNPE) [39] and DLPP/MMC [40] are also proposed to solve the SSS problem of NPE and DLPP, respectively.

Recently, matrix exponential is introduced to solve the SSS problem in the manifold learning field and some methods have been proposed, such as exponential neighborhood preserving embedding (ENPE) [41]; exponential LPP (ELPP) [42]; exponential DLPP (EDLPP) [43], [44]; exponential LDE (ELDE) [45]; exponential semisupervised discriminant embedding (ESDE) [46]; exponential neighborhood preserving discriminant embedding (ENPDE) [47]; etc. In [48], a general exponential framework has been presented. Although these methods used the matrix exponential differently, in essence, they replaced the matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ in (1) with the matrix exponentials $\exp(\boldsymbol{S}_1)$ and $\exp(\boldsymbol{S}_2)$, thereby avoiding the singularity of the matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$.

In this article, from the perspective of unsupervised and supervised, the existing dimensionality reduction methods are first summarized, then a unified criterion of these methods is established. Under the unified criterion, a general matrix function dimensionality reduction framework is proposed. This framework is configurable, that is, one can take suitable functions to construct the matrix functions and then to construct a matrix transformation framework. This idea comes from solving the SSS problem of dimensionality reduction methods. However, the framework is not confined to solving the SSS problem. When suitable functions are chosen, the method is also beneficial to the pattern classification. We discuss how to choose suitable functions from two aspects: 1) solving the SSS problem and 2) improving pattern classification ability. Besides, under the matrix transformation framework, the above regularization method and the matrix exponential method may be viewed as special instances of this framework.

As an extension and application of the proposed framework, with the "inverse hyperbolic tangent function" and the "linear function," a new matrix transformation framework is derived and applied to the popular manifold learning methods. Because LPP has been paid more attention to by researchers, LPP, DLPP, function LPP (FLPP), and function DLPP (FDLPP) are taken as examples to study, where FLPP and FDLPP are derived from the proposed framework. The new FLPP and FDLPP methods can avoid the SSS problem. Moreover, compared with the existing methods, they have better pattern classification ability. The experimental results validate that the new framework is an efficient method.

The remainder of this article proceeds as follows. In Section II, we summarize the criteria for the existing dimensionality reduction methods. In Section III, we establish a general matrix function dimensionality reduction framework and discuss how to choose the appropriate function. In Section IV, the relationship between this framework and the existing methods is discussed. In Section V, as an extension, a new matrix function dimensionality reduction framework is presented. We make experiments to evaluate the proposed method in Section VI and finally conclude this article in Section VII.

II. Dimensionality Reduction Review on Graph Embedding View

Let $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m\}$ be m original data in R^n space. In the past few years, many subspace learning methods aim to seek an optimal transformation to map the n-dimensional data point $\boldsymbol{x}_i$ onto a d-dimensional ($d \ll n$) data point $\boldsymbol{y}_i$. In [32], a general graph embedding framework is proposed to unify the subspace learning methods. In the view of graph

TABLE I
DEFINITION OF THE MATRICES $\boldsymbol{S}_W$, $\boldsymbol{S}_D$, $\boldsymbol{W}$, AND $\boldsymbol{D}$

Methods	$\boldsymbol{S}_W$	$\boldsymbol{S}_D$	$\boldsymbol{W}$ and $\boldsymbol{D}$
LPP	$\boldsymbol{XWX}^T$	$\boldsymbol{XDX}^T$	$\boldsymbol{W}=(\boldsymbol{W}_{ij})$, $D_{ii}=\sum_j W_{ij}$,where $W_{ij}=\exp\left(-\left\|x_i-x_j\right\|^2/t\right)$, if $\boldsymbol{x}_i\in N_k(\boldsymbol{x}_j)or\,\boldsymbol{x}_j\in N_k(\boldsymbol{x}_i)$, otherwise =0
NPE	$\boldsymbol{XWX}^T$	$\boldsymbol{XX}^T$	$\boldsymbol{W}=\boldsymbol{M}+\boldsymbol{M}^T+\boldsymbol{M}^T\boldsymbol{M}$, $\boldsymbol{D}=\boldsymbol{I}$, where $\boldsymbol{I}$ is the identity matrix $\boldsymbol{M}=(M_{ij})$, M_{ij} is the weight coefficients that best reconstruct $\boldsymbol{x}_i$ from its neighbors $\boldsymbol{x}_j$
IsoP	$\boldsymbol{X}\tau(\boldsymbol{D}_G)\boldsymbol{X}^T$	$\boldsymbol{XX}^T$	$\boldsymbol{W}=\tau(\boldsymbol{D}_G)$, $\boldsymbol{D}=\boldsymbol{I}$, $\tau(\boldsymbol{D}_G)=-\boldsymbol{H}_\tau\boldsymbol{S}\boldsymbol{H}_\tau/2$, $\boldsymbol{H}_\tau=\boldsymbol{I}-\frac{1}{m}\boldsymbol{e}\boldsymbol{e}^T$, where $\boldsymbol{I}$ is the identity matrix, $\boldsymbol{e}=(1,1,\cdots,1)^T$, $\boldsymbol{S}=(S_{ij}),S_{ij}=d_{ij}^2$, d_{ij} is the distance between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$

embedding, a graph $\boldsymbol{G}=\{\boldsymbol{X},\boldsymbol{W}\}$ is defined and used to characterize the dataset, where $\boldsymbol{X}$ is the vertex set of the graph $\boldsymbol{G}$, $\boldsymbol{W}=(W_{ij})$ is a similarity matrix, and W_{ij} measures the similarity of a pair of vertices i and j. The graph-preserving criterion is

$$\boldsymbol{y}^*=\arg\min_{\boldsymbol{y}^T\boldsymbol{B}\boldsymbol{y}=d}\sum_{i\neq j}\left\|\boldsymbol{y}_i-\boldsymbol{y}_j\right\|^2W_{ij}=\arg\min_{\boldsymbol{y}^T\boldsymbol{B}\boldsymbol{y}=d}\boldsymbol{y}^T\boldsymbol{L}\boldsymbol{y} \tag{2}$$

where $\boldsymbol{B}$ is a constraint matrix, d is a constant, $\boldsymbol{L}=\boldsymbol{D}-\boldsymbol{W}$ is a Laplacian matrix, $\boldsymbol{D}$ is a diagonal matrix, and $D_{ii}=\sum_j W_{ij}$.

A. Unsupervised manifold learning

In the case of unsupervised learning, many linearization methods, for example, LPP, NPE, and ISOMAP, have been developed based on the classical manifold learning algorithms. In [37], [49], and [50], based on the graph embedding framework, a method, called a linear extension of graph embedding (LGE), is developed to unify these linearization methods. Let $y_i=\boldsymbol{u}^T\boldsymbol{x}_i$ be the linear mapping from the n-dimensional original data $\boldsymbol{x}_i\in R^n$ onto the d-dimensional data $\boldsymbol{y}_i\in R^d$ $(i=1,2,\ldots,m)$, and denoting $\boldsymbol{y}=(y_1,y_2,\ldots,y_m)^T$, one has $\boldsymbol{y}=\boldsymbol{X}^T\boldsymbol{u}$. Under the LGE framework, these methods may be formulated as the following criterion:

$$\boldsymbol{u}^*=\arg\max\frac{\boldsymbol{y}^T\boldsymbol{W}\boldsymbol{y}}{\boldsymbol{y}^T\boldsymbol{D}\boldsymbol{y}}=\arg\max\frac{\boldsymbol{u}^T\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^T\boldsymbol{u}}{\boldsymbol{u}^T\boldsymbol{X}\boldsymbol{D}\boldsymbol{X}^T\boldsymbol{u}}. \tag{3}$$

Generally, an optimal projection matrix $\boldsymbol{U}$ is sought to make the mapping, namely, $\boldsymbol{y}_i=\boldsymbol{U}^T\boldsymbol{x}_i$. Thus, the criterion of these methods can be reformulated as

$$\boldsymbol{J}(\boldsymbol{U})=\arg\max_U\frac{\left|\boldsymbol{U}^T\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^T\boldsymbol{U}\right|}{\left|\boldsymbol{U}^T\boldsymbol{X}\boldsymbol{D}\boldsymbol{X}^T\boldsymbol{U}\right|}. \tag{4}$$

The detail about LGE may be referred to [37], [49], and [50].

Denote $\boldsymbol{S}_W=\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^T$ and $\boldsymbol{S}_D=\boldsymbol{X}\boldsymbol{D}\boldsymbol{X}^T$, then (4) becomes

$$\boldsymbol{J}(\boldsymbol{U})=\arg\max_U\frac{\left|\boldsymbol{U}^T\boldsymbol{S}_W\boldsymbol{U}\right|}{\left|\boldsymbol{U}^T\boldsymbol{S}_D\boldsymbol{U}\right|}. \tag{5}$$

LPP, NPE, and ISOMAP can be unified into the framework (5) but the forms of the matrices $\boldsymbol{S}_W$, $\boldsymbol{S}_D$, and $\boldsymbol{W}$ are different. Table I lists the forms of the matrices for LPP, NPE, and ISOMAP.

B. Supervised Manifold Learning

LPP, NPE, ISOMAP, and LLTSA methods are unsupervised learning methods. Some supervised manifold learning methods, which use the category information of the samples during training, also are developed. Representative ones are of LDE [28], DLPP [29], NPDE [31], MFA [32], etc. They can be regarded as an extension of the linearized version of manifold learning to supervised learning. These methods can also be unified into the graph embedding framework, that is, (2) [51].

With the category information of samples, these methods construct the intraclass weight matrix $\boldsymbol{W}^w$ based on the intraclass samples, and the interclass weight matrix $\boldsymbol{W}^b$ based on the interclass samples, respectively. These methods are also to make the mapping from the n-dimensional original data $\boldsymbol{x}_i\in R^n$ onto the d-dimensional data $\boldsymbol{y}_i\in R^d$ $(d\ll n)$, namely, $\boldsymbol{y}_i=\boldsymbol{U}^T\boldsymbol{x}_i$, and then to seek the optimal projection matrix $\boldsymbol{U}$. The derivation of the matrix $\boldsymbol{U}$ is based on Fisher's criterion. Thus, despite different motivations, the criterion of these methods can be unified as

$$\boldsymbol{J}(\boldsymbol{U})=\arg\max_U\frac{\left|\boldsymbol{U}^T\boldsymbol{X}\boldsymbol{L}_B\boldsymbol{X}^T\boldsymbol{U}\right|}{\left|\boldsymbol{U}^T\boldsymbol{X}\boldsymbol{L}_W\boldsymbol{X}^T\boldsymbol{U}\right|}. \tag{6}$$

We denote $\boldsymbol{S}_B=\boldsymbol{X}\boldsymbol{L}_B\boldsymbol{X}^T$ and $\boldsymbol{S}_W=\boldsymbol{X}\boldsymbol{L}_W\boldsymbol{X}^T$, then (6) becomes

$$\boldsymbol{J}(\boldsymbol{U})=\arg\max_U\frac{\left|\boldsymbol{U}^T\boldsymbol{S}_B\boldsymbol{U}\right|}{\left|\boldsymbol{U}^T\boldsymbol{S}_W\boldsymbol{U}\right|} \tag{7}$$

where the matrices $\boldsymbol{S}_B$, $\boldsymbol{S}_W$, $\boldsymbol{L}_B$, and $\boldsymbol{L}_W$ have different forms for the different methods. Table II lists the forms of the matrices for LDE, DLPP, and MFA.

III. GENERAL MATRIX FUNCTION DIMENSIONALITY REDUCTION FRAMEWORK

A. Unified Framework for Manifold Learning

In Section II, in the unsupervised and supervised cases, the criteria of dimensionality reduction methods may be reformulated as (5) and (7), respectively. These methods are to find an optimal matrix $\boldsymbol{U}$, then to maximize the following criterion function:

$$\boldsymbol{J}(\boldsymbol{U})=\arg\max_U\frac{\left|\boldsymbol{U}^T\boldsymbol{S}_1\boldsymbol{U}\right|}{\left|\boldsymbol{U}^T\boldsymbol{S}_2\boldsymbol{U}\right|} \tag{8}$$

and then to make a linear mapping from the n-dimensional original data $\boldsymbol{x}_i\in R^n$ to d-dimensional data $\boldsymbol{y}_i\in R^d$, namely, $\boldsymbol{y}_i=\boldsymbol{U}^T\boldsymbol{x}_i$, where, for the popular dimensionality reduction methods, the matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ may be taken from Tables I and II according to the desired method. The optimal

TABLE II
DEFINITION OF THE MATRICES $\boldsymbol{S}_B$, $\boldsymbol{S}_W$, $\boldsymbol{L}_B$, AND $\boldsymbol{L}_W$

Methods	$\boldsymbol{S}_B$	$\boldsymbol{S}_W$	$\boldsymbol{L}_B$ and $\boldsymbol{L}_W$
LDE	$\boldsymbol{X}\boldsymbol{L}_b\boldsymbol{X}^T$	$\boldsymbol{X}\boldsymbol{L}_w\boldsymbol{X}^T$	$\boldsymbol{L}_b=\boldsymbol{D}^b-\boldsymbol{W}^b$, $D^b_{ii}=\sum_j W^b_{ij}$, where $W^b_{ij}=1$, if $\boldsymbol{x}_i\in N_b(\boldsymbol{x}_j)or\,\boldsymbol{x}_j\in N_b(\boldsymbol{x}_i)$, otherwise =0 $\boldsymbol{L}_w=\boldsymbol{D}^w-\boldsymbol{W}^w$, $D^w_{ii}=\sum_j W^w_{ij}$, where $W^w_{ij}=sim(\boldsymbol{x}_i,\boldsymbol{x}_j)$, if $\boldsymbol{x}_i\in N_w(\boldsymbol{x}_j)or\,\boldsymbol{x}_j\in N_w(\boldsymbol{x}_i)$, otherwise = 0
DLPP	$\boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^T$	$\overline{\boldsymbol{X}}\boldsymbol{H}\overline{\boldsymbol{X}}^T$	$\boldsymbol{L}=\boldsymbol{D}-\boldsymbol{W}$, $\boldsymbol{D}=diag(\boldsymbol{D}_1,\boldsymbol{D}_2,\cdots,\boldsymbol{D}_C)$, $D^c_{ii}=\sum_j W^c_{ij}$, $\boldsymbol{W}=diag(\boldsymbol{W}_1,\boldsymbol{W}_2,\cdots,\boldsymbol{W}_C)$, where $\boldsymbol{W}_c$ is the weight matrix between any two samples in cth class, $W^c_{ij}=\exp\left(-\|\boldsymbol{x}_i-\boldsymbol{x}_j\|^2/t\right)$. $\boldsymbol{H}=\boldsymbol{E}-\boldsymbol{B}$, $E_{ii}=\sum_j B_{ij}$, $B_{ij}=\exp\left(-\|\boldsymbol{f}_i-\boldsymbol{f}_j\|^2/t\right)$, $\overline{\boldsymbol{X}}=\left[\overline{\boldsymbol{x}_1},\overline{\boldsymbol{x}_2},...,\overline{\boldsymbol{x}_C}\right]$, where $\boldsymbol{f}_i$, $\overline{\boldsymbol{x}_i}$ is the mean vector of the ith class
MFA	$\boldsymbol{X}\boldsymbol{L}_b\boldsymbol{X}^T$	$\boldsymbol{X}\boldsymbol{L}_w\boldsymbol{X}^T$	$\boldsymbol{L}_b=\boldsymbol{D}^b-\boldsymbol{W}^b$, $\boldsymbol{D}^b_{ii}=\sum_j \boldsymbol{W}^b_{ij}$, where $W^b_{ij}=1$, if $(i,j)\in P_{k_2}(c_i)or(i,j)\in P_{k_2}(c_j)$, otherwise = 0 $\boldsymbol{L}_w=\boldsymbol{D}^w-\boldsymbol{W}^w$, $\boldsymbol{D}^w_{ii}=\sum_j \boldsymbol{W}^w_{ij}$, where $W^w_{ij}=1$, if $\boldsymbol{x}_i\in N^+_{k_2}(\boldsymbol{x}_j)or\,\boldsymbol{x}_j\in N^+_{k_2}(\boldsymbol{x}_i)$, otherwise = 0

projection matrix $\boldsymbol{U}$ may be obtained from the following generalized eigenproblem:

$$\boldsymbol{S}_1\boldsymbol{u}=\lambda\boldsymbol{S}_2\boldsymbol{u}.$$

In this view, (8) can be viewed as a unified framework of these methods. When the SSS problem occurs, the matrix $\boldsymbol{S}_2$ is singular and the above eigenproblem is unsolvable [41]–[48].

B. Matrix Function and Its Eigensystem

In this article, the matrix function is the theoretical basis of the proposed matrix function dimensionality reduction framework. So, the definition and eigensystem of the matrix function are presented in this section. In mathematics, a matrix function may be viewed as a function that maps a matrix to another matrix. Roughly speaking, the definition of the matrix function is described as follows.

Definition 1 [52]: Let $\boldsymbol{A}$ be an n-order square matrix. If the scalar function $f(x)$ is defined on $\lambda(\boldsymbol{A})$, where $\lambda(\boldsymbol{A})$ is the eigenvalue set of the matrix $\boldsymbol{A}$, then $f(\boldsymbol{A})$ is defined by replacing "$\boldsymbol{A}$" with "x" in the "formula" of $f(x)$.

Theorem 1 [52]: Let $\boldsymbol{A}$ be an n-order diagonalizable square matrix, that is, there exists an n-order matrix $\boldsymbol{V}$, subject to $A=\boldsymbol{V}\text{diag}(\lambda_1,\lambda_2,\ldots,\lambda_n)\boldsymbol{V}^{-1}$. Let $f(\boldsymbol{A})$ be a matrix function, then

$$f(\boldsymbol{A})=\boldsymbol{V}\text{diag}(f(\lambda_1),f(\lambda_2),\ldots,f(\lambda_n))\boldsymbol{V}^{-1}. \tag{9}$$

So, by Theorem 1, we may obtain the following theorem about the eigensystem of a matrix function $f(\boldsymbol{A})$.

Theorem 2: Let $\boldsymbol{A}$ be an n-order real symmetric square matrix and $f(\boldsymbol{A})$ be a matrix function with relation to $\boldsymbol{A}$, if

$$\boldsymbol{A}\boldsymbol{v}_i=\lambda_i\boldsymbol{v}_i(i=1,2,\ldots,n)$$

where λ_i is the eigenvalue of the matrix $\boldsymbol{A}$, and $\boldsymbol{v}_i$ is the eigenvector of $\boldsymbol{A}$ belonging to λ_i, then one has

$$f(\boldsymbol{A})\boldsymbol{v}_i=f(\lambda_i)\boldsymbol{v}_i(i=1,2,\ldots,n) \tag{10}$$

that is, $f(\lambda_i)$ is the eigenvalue of the matrix function $f(\boldsymbol{A})$, and $\boldsymbol{v}_i$ is also the eigenvector of $f(\boldsymbol{A})$ belonging to $f(\lambda_i)$.

Proof: Because the matrix $\boldsymbol{A}$ is a real symmetric square matrix, it can be diagonalized. Suppose that the matrix $\boldsymbol{A}$ is diagonalized as $\boldsymbol{V}^{-1}\boldsymbol{A}\boldsymbol{V}=\text{diag}(\lambda_1,\lambda_2,\ldots,\lambda_n)$, where λ_i is the eigenvalue of $\boldsymbol{A}$, $\boldsymbol{v}_i$ is the corresponding eigenvector belonging to the eigenvalue λ_i, and the matrix $\boldsymbol{V}$ is composed of the eigenvectors $\boldsymbol{v}_i(i=1,2,\ldots,n)$, $\boldsymbol{V}=[\boldsymbol{v}_1,\boldsymbol{v}_2,\ldots,\boldsymbol{v}_n]$. Then, the matrix $\boldsymbol{A}$ may be rewritten as

$$\boldsymbol{A}=\boldsymbol{V}\text{diag}(\lambda_1,\lambda_2,\ldots,\lambda_n)\boldsymbol{V}^{-1}.$$

By the above theorem (Theorem 1), the matrix function $f(\boldsymbol{A})$ may be written as the following form:

$$f(\boldsymbol{A})=\boldsymbol{V}\text{diag}(f(\lambda_1),f(\lambda_2),\ldots,f(\lambda_n))\boldsymbol{V}^{-1}.$$

Then, one has

$$f(\boldsymbol{A})\boldsymbol{v}_i=f(\lambda)_i\boldsymbol{v}_i(i=1,2,\ldots,n).$$

■

C. General Matrix Function Dimensionality Reduction Framework

Let $f(x)$ and $g(x)$ denote two scalar functions, the two functions can be the same or different. With the functions $f(x)$ and $g(x)$, the corresponding matrix functions of the matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ in (8) are $f(\boldsymbol{S}_1)$ and $g(\boldsymbol{S}_2)$. In fact, the matrix function may be viewed as a nonlinear mapping function $\Theta:\mathbb{R}^{n\times n}\to\mathbb{R}^{n\times n}$

$$\boldsymbol{S}_1\to\Theta(\boldsymbol{S}_1)=f(\boldsymbol{S}_1) \tag{11}$$

$$\boldsymbol{S}_2\to\Theta(\boldsymbol{S}_2)=g(\boldsymbol{S}_2). \tag{12}$$

Note that the matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ are composed of the original samples, the above mapping transforms the matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ from the original sample space into $f(\boldsymbol{S}_1)$ and $g(\boldsymbol{S}_2)$ in a new space. Then, in such a new space, one has the criterion function

$$J(\boldsymbol{U})=\arg\max_U\frac{\left|\boldsymbol{U}^Tf(\boldsymbol{S}_1)\boldsymbol{U}\right|}{\left|\boldsymbol{U}^Tg(\boldsymbol{S}_2)\boldsymbol{U}\right|}. \tag{13}$$

With some algebra transformation, the optimal mapping transformation matrix $\boldsymbol{U}$ may be obtained by composing the eigenvectors belonging to the maximum eigenvalues of the following eigenproblem:

$$f(\boldsymbol{S}_1)\boldsymbol{u}=\lambda g(\boldsymbol{S}_2)\boldsymbol{u}. \tag{14}$$

We may select the two functions $f(x)$ and $g(x)$ that satisfy $f(x) \neq 0$ when $x = 0$ and $g(x) \neq 0$ when $x = 0$. Let λ_i^1 be any eigenvalue of the matrix $\boldsymbol{S}_1$, and λ_i^2 be any eigenvalue of the matrix $\boldsymbol{S}_2$. Note that the matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ are semipositive definite. By Theorem 2, the eigenvalue of the matrix function $f(\boldsymbol{S}_1)$ has the form $f(\lambda_i^1)$ and $f(\lambda_i^1) \neq 0$; similarly, the eigenvalue of the matrix function $g(\boldsymbol{S}_2)$ has the form $g(\lambda_i^2)$ and $g(\lambda_i^2) \neq 0$. When the SSS problem occurs, the matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ are singular, that is, there are some zero eigenvalues. But, after the matrix transformation as in (11) and (12), in such a new space, because the eigenvalues of $f(\boldsymbol{S}_1)$ and $g(\boldsymbol{S}_2)$ are nonzero, $f(\boldsymbol{S}_1)$ and $g(\boldsymbol{S}_2)$ are always nonsingular, and then (14) is always solvable. Thus, the SSS problem is addressed.

In (13), $f(x)$ and $g(x)$ are configurable. That is, one can take two suitable functions to construct the corresponding matrix functions and then to construct a matrix transformation framework. Thus, a general matrix function dimensionality reduction framework is presented. With the matrix transformation framework, linear manifold learning (unsupervised/supervised) may be transformed into a new space, and then the corresponding dimensionality reduction criterion is used in such a new space.

Fig. 1 illustrates the main idea of the general matrix function dimensionality reduction framework.

D. Selection of Matrix Function and Analysis

The motivation to select suitable functions $f(x)$ and $g(x)$ comes from two aspects. First, they can address the SSS problem. Second, it is expected that two suitable functions can make the framework having better performance, for example, better pattern classification ability.

1) Solving the SSS Problem: From the above section, the matrix function dimensionality reduction framework sets the optical projection axes $\boldsymbol{u}$'s to the eigenvectors of (14). To solve the SSS problem, we may choose a function $g(x)$ such that $g(\lambda_i^2) \neq 0$. Therefore, even if the matrix $\boldsymbol{S}_2$ is singular, the matrix function $g(\boldsymbol{S}_2)$ is always nonsingular. Then, (14) is always solvable and the SSS problem is avoided.

A common way to solve SSS problems is to reduce the dimensions of the original sample using the PCA method, but due to the dimensionality reduction of the PCA method, some important information is discarded. Under the matrix function dimensionality reduction framework, these manifold learning methods can be executed directly without using PCA for dimensionality reduction. Then, all the information that are contained in $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ can be extracted. So, we can believe that the methods under the matrix transformation framework have more discriminant ability than the classical methods.

2) Distance Diffusion Mapping: For the supervised algorithms, including DLPP, LDE, NPDE, MFA, etc., these methods were proposed with different motivations, but the local interclass and intraclass distance in the sample space may be uniformly formulated as

$$d_b = \frac{1}{2}\sum_{i,j}\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 W_{ij}^b = tr\left(\boldsymbol{X}\boldsymbol{L}_b\boldsymbol{X}^T\right) = tr(\boldsymbol{S}_B) \quad (15)$$

$$d_w = \frac{1}{2}\sum_{i,j}\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 W_{ij}^w = tr\left(\boldsymbol{X}\boldsymbol{L}_W\boldsymbol{X}^T\right) = tr(\boldsymbol{S}_W) \quad (16)$$

where $tr(\cdot)$ denotes the trace of a matrix.

Let $\lambda_{bi}, \lambda_{wi} (i = 1, 2, \ldots, n)$ be the eigenvalues of the matrices $\boldsymbol{S}_B$ and $\boldsymbol{S}_W$, respectively, the above two distances may be written as

$$d_b = tr(\boldsymbol{S}_B) = \lambda_{b1} + \lambda_{b2} + \cdots + \lambda_{bn} \quad (17)$$

$$d_w = tr(\boldsymbol{S}_W) = \lambda_{w1} + \lambda_{w2} + \cdots + \lambda_{wn}. \quad (18)$$

From (17) and (18), the eigenvalues measure the interclass and intraclass distance. By the mapping (11) and (12), the matrices $\boldsymbol{S}_B$ and $\boldsymbol{S}_W$ are mapping into the matrices $f(\boldsymbol{S}_B)$ and $g(\boldsymbol{S}_W)$ in a new space. By Theorem 2, the local interclass and intraclass distances in such new space become

$$d_b^f = tr(f(\boldsymbol{S}_B)) = f(\lambda_{b1}) + f(\lambda_{b2}) + \cdots + f(\lambda_{bn}) \quad (19)$$

$$d_w^f = tr(g(\boldsymbol{S}_W)) = g(\lambda_{w1}) + g(\lambda_{w2}) + \cdots + g(\lambda_{wn}). \quad (20)$$

In general, the distance of local interclasses is larger than that of the local intraclasses, that is, for most of the eigenvalues in (17) and (18), one has the inequality $\lambda_{bi} > \lambda_{wi}$. In patter classification, we always want to make the distance of interclass samples as large as possible, and the distance of intraclass samples as close as possible simultaneously. So, we can select $f(x)$ such that $f(\lambda_{bi}) > \lambda_{bi}$, and a function $g(x)$ such that $g(\lambda_{wi}) < \lambda_{wi}$ or $g(\lambda_{wi}) \approx \lambda_{wi}$. Thus, one has

$$\frac{f(\lambda_{bi})}{g(\lambda_{wi})} > \frac{\lambda_{bi}}{\lambda_{wi}}.$$

Then, the following inequality holds:

$$\frac{d_b^f}{d_w^f} > \frac{d_b}{d_w}.$$

With the matrix function transformation (11) and (12), the original sample space is transformed into a new sample space. If the functions $f(x)$ and $g(x)$ are properly selected, in the new space, the ratio of the distance of local interclasses to the distance of the local intraclasses is enlarged. So, the discriminating property is largely emphasized and then the separability of the pattern will be improved.

IV. Relationship With the Existing Methods

In this section, we will discuss the relationship of the proposed general framework with the existing methods. We can find that the regularization method and the matrix exponential method are special cases of this method and can be integrated into the matrix transformation framework.

The regularization method is a common way to deal with the SSS problem. The criterion function of this method may be formulated as

$$\boldsymbol{J}(\boldsymbol{U}) = \arg\max_{U} \frac{\left|\boldsymbol{U}^T\boldsymbol{S}_1\boldsymbol{U}\right|}{\left|\boldsymbol{U}^T(r\boldsymbol{I} + \boldsymbol{S}_2)\boldsymbol{U}\right|} \quad (21)$$

where r is a regularization parameter.

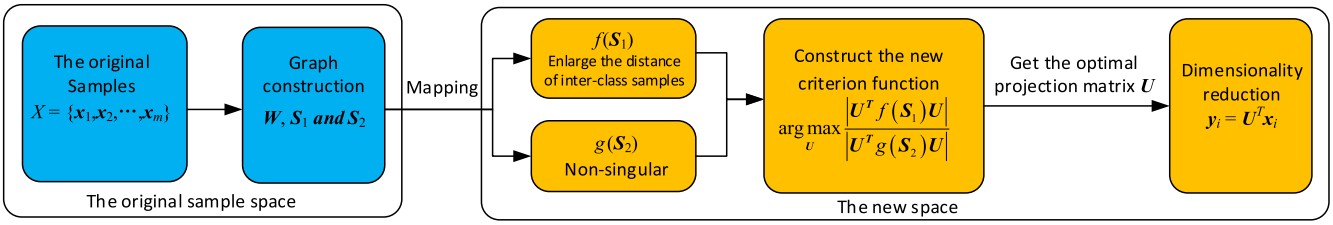

Fig. 1. Flowchart of the general matrix function dimensionality reduction framework.

Let $f(x) = x$ and $g(x) = r + x(r > 0)$, their matrix function are $f(\boldsymbol{S}_1) = \boldsymbol{S}_1$ and $g(\boldsymbol{S}_2) = r\boldsymbol{I} + \boldsymbol{S}_2$. In this view, the above criterion (21) is unified into (13). That is to say, the regularization method may be viewed as a special instance derived from the matrix function dimensionality reduction framework (13).

The matrix exponential method has recently proposed to address the SSS problem. It replaced the matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ in (8) with the matrix exponentials $\exp(\boldsymbol{S}_1)$ and $\exp(\boldsymbol{S}_2)$. In fact, it may also be derived from the framework (13). Let the function $f(x) = g(x) = e^x$, then one has the matrix functions $f(\boldsymbol{S}_1) = \exp(\boldsymbol{S}_1)$ and $g(\boldsymbol{S}_2) = \exp(\boldsymbol{S}_2)$. Substituting the matrix functions $\exp(\boldsymbol{S}_1)$ and $\exp(\boldsymbol{S}_2)$ into (13), one has

$$J(\boldsymbol{U}) = \arg\max_{U} \frac{\left|\boldsymbol{U}^T \exp(\boldsymbol{S}_1)\boldsymbol{U}\right|}{\left|\boldsymbol{U}^T \exp(\boldsymbol{S}_2)\boldsymbol{U}\right|}. \tag{22}$$

Equation (22) is the criterion function of the matrix exponential methods, as in [43], [44], [47], etc.

Therefore, for the popular methods, such as LPP, NPE, DLPP, LDE, and MFA, if the two matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ in (21) and (22) are replaced with the matrices in Tables I and II according to desired methods, we may obtain the corresponding regularization or exponential version of the above methods.

V. Extension and Application of the Matrix Function Dimensionality Reduction Framework

A. New Matrix Function Framework

Based on Section III, if appropriate functions are used, the proposed framework can be used to design new dimensionality reduction methods. As an extension and application of the proposed framework, a new dimensionality reduction framework is presented in this section.

In deep learning, the hyperbolic tangent function $y = \tanh(x)$ is often used as an activation function of the neural network. In this article, the "inverse hyperbolic tangent function" $y = \operatorname{artanh}(x)$ is used. Let $f(x) = 1 + \operatorname{artanh}(x)$, $g(x)$ be a linear function, that is, $g(x) = r + x(r > 0)$, and let $\boldsymbol{A}$ be a square matrix, the matrix functions of $f(x)$ and $g(x)$ are

$$f(\boldsymbol{A}) = \boldsymbol{I} + \operatorname{artanh}(\boldsymbol{A}) \tag{23}$$

$$g(\boldsymbol{A}) = r\boldsymbol{I} + \boldsymbol{A}. \tag{24}$$

Then, the matrix function framework (13) becomes

$$J(\boldsymbol{U}) = \arg\max_{U} \frac{\left|\boldsymbol{U}^T(\boldsymbol{I} + \operatorname{artanh}(\boldsymbol{S}_1))\boldsymbol{U}\right|}{\left|\boldsymbol{U}^T(r\boldsymbol{I} + \boldsymbol{S}_2)\boldsymbol{U}\right|}. \tag{25}$$

The optimal mapping transformation vector $\boldsymbol{u}$'s are the eigenvectors belonging to the maximum eigenvalue of the generalized eigenproblem

$$(\boldsymbol{I} + \operatorname{artanh}(\boldsymbol{S}_1))\boldsymbol{u} = \lambda(r\boldsymbol{I} + \boldsymbol{S}_2)\boldsymbol{u}. \tag{26}$$

Under the framework (25), the corresponding criterion functions may be obtained for different dimensionality reduction methods. So, for the popular methods, such as LPP, NPE, LDE, DLPP, NPDE, and MFA, the criteria under the matrix transformation framework may be obtained. For convenience, these methods are called FLPP, FNPE, FLDE, FDLPP, FNPDE, and FMFA, respectively. The initial "F" means "Function."

B. Analysis

Why are the "inverse hyperbolic tangent function" $f(x) = 1 + \operatorname{artanh}(x)$ and "linear function" $g(x) = r + x$ selected to make the matrix transformation ?

1) Solving the SSS Problem: Let λ_{S_2} be any eigenvalue of the matrix $\boldsymbol{S}_2$. Note that the matrix $\boldsymbol{S}_2$ is semipositive definite, one has $\lambda_{S_2} \geq 0$. From Theorem 2, the eigenvalue of the matrix function $g(\boldsymbol{A}) = r\boldsymbol{I} + \boldsymbol{A}$ has the form $r + \lambda_{S_2}$. One has $r + \lambda_{S_2} \geq r$, so the matrix function $g(\boldsymbol{A}) = r\boldsymbol{I} + \boldsymbol{A}$ is nonsingular. Thus, (26) is always solvable even if the SSS problem occurs.

2) Distance Diffusion Mapping: In this section, we discuss the distance diffusion effect of the proposed matrix function framework for the supervised methods. It is necessary to note that the matrices $\boldsymbol{S}_B$ and $\boldsymbol{S}_W$ of the supervised methods are normalized with their norms. One reason for this processing is that it will avoid large values of $\exp(\boldsymbol{S}_B)$ and $\exp(\boldsymbol{S}_W)$, and another reason is that after normalization, the eigenvalues of the matrices $\boldsymbol{S}_B$ and $\boldsymbol{S}_W$ are in the interval [0,1], and thus within the domain of the inverse hyperbolic tangent function. Such processing is equivalent to shrinking the eigenvalue to the interval [0, 1]. Because we care about the ratio $\lambda_{b_i}/\lambda_{w_i}$, which evaluates the ratio of the distance of interclass samples and the distance of intraclass samples, it has nothing to do with the absolute value of the eigenvalues.

The curves of the function $f(x) = 1 + \operatorname{artanh}(x)$ and the function $y = x$ have been presented in Fig. 2. From Fig. 2, the domain of the function $f(x) = 1 + \operatorname{artanh}(x)$ is the interval [0, 1]. When $x \to 1$, $f(x) \to \infty$. So, the larger the eigenvalue λ, the larger its function value $f(\lambda)$.

For the supervised methods, under the proposed framework, the matrices $\boldsymbol{S}_B$ and $\boldsymbol{S}_W$ are mapped into a new space, that is

$$\boldsymbol{S}_B \to f(\boldsymbol{S}_B) = \boldsymbol{I} + \operatorname{artanh}(\boldsymbol{S}_B) \tag{27}$$

$$\boldsymbol{S}_W \to g(\boldsymbol{S}_W) = r\boldsymbol{I} + \boldsymbol{S}_W. \tag{28}$$

Let $\lambda_{bi}, \lambda_{wi} (i = 1, 2, \ldots, n)$ be the eigenvalues of the matrices $\boldsymbol{S}_B$ and $\boldsymbol{S}_W$, respectively, and for most of the eigenvalues, one

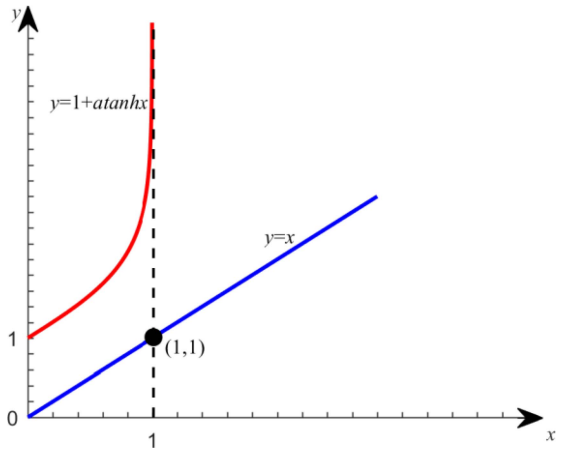

Fig. 2. Curves of the inverse hyperbolic tangent function and linear function.

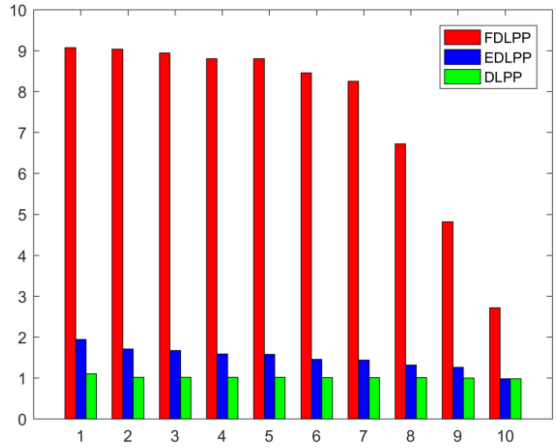

Fig. 3. Comparison of three proportional values.

has $\lambda_{b_i} > \lambda_{w_i}$. By Theorem 2, the eigenvalues of $f(\boldsymbol{S}_B) = \boldsymbol{I} + \text{artanh}(\boldsymbol{S}_B)$ and $g(\boldsymbol{S}_W) = r\boldsymbol{I} + \boldsymbol{S}_W$ have the form $1 + \text{artanh}(\lambda_{b_i})$ and $r + \lambda_{w_i}$. From Fig. 2, one has the inequality $1 + \text{artanh}(\lambda_{b_i}) > \lambda_{b_i}$, and $r + \lambda_{w_i} \approx \lambda_{w_i}$ (generally, r take a little positive value), so the following inequality holds:

$$\frac{1 + \text{artanh}(\lambda_{bi})}{r + \lambda_{wi}} > \frac{\lambda_{bi}}{\lambda_{wi}}. \tag{29}$$

So, the functions $f(x) = 1 + \text{artanh}(x)$ and $g(x) = r + x$ have the distance diffusion effect described in Section III-D.

As discussed in Section IV, the matrix exponential method is a special case of the general matrix function dimensionality reduction framework, so the matrix exponential method also has the function of distance diffusion mapping. However, with the mapping of the exponential function $h(x) = e^x$, the eigenvalues λ_{bi} and λ_{wi} are enlarged to $e^{\lambda_{bi}}$ and $e^{\lambda_{wi}}$. With the mapping of $f(x) = 1 + \text{artanh}(x)$ and $g(x) = r + x$, the eigenvalue λ_{bi} is enlarged to $1 + \text{artanh}(\lambda_{bi})$, and note that $1 + \text{artanh}(\lambda_{bi}) \approx e^{\lambda_{b_i}}$ if $0 < \lambda_{bi} < 1$. Simultaneously, λ_{wi} is mapped into $r + \lambda_{w_i}$, one has $r + \lambda_{w_i} \approx \lambda_{w_i}$ and $r + \lambda_{w_i} < e^{\lambda_{w_i}}$. So one has the following inequality:

$$\frac{1 + \text{artanh}(\lambda_{bi})}{r + \lambda_{wi}} > \frac{e^{\lambda_{bi}}}{e^{\lambda_{wi}}}. \tag{30}$$

Thus, from (29) and (30), the proposed matrix function framework has a distance diffusion effect and shows a more strong diffusion ability than the existing method.

To illustrate this, with DLPP, EDLPP, and FDLPP methods, an experiment is made in the Georgia Tech face database. In the experiment, the three proportional expressions in (29) and (30) are calculated, respectively, with the corresponding method and compared in Fig. 3. In this figure, the red line represents the proportional value $(1 + \text{artanh}(\lambda_{bi}))/(r + \lambda_{wi})$ from the FDLPP method, the blue line represents the proportional value $e^{\lambda_{bi}}/e^{\lambda_{wi}}$ from the EDLPP method, and the green line represents the proportional value $\lambda_{bi}/\lambda_{wi}$ from the DLPP method, where the first ten largest eigenvalues of the locality preserving scatter matrices are taken to calculate the proportional values.

As can be seen from Fig. 3, the diffusion scale of the proposed matrix function framework is much larger than that of the other two methods. Moreover, from the above analysis, the distance between different class samples is enlarged by the proposed method. Thus, the proposed matrix function framework will show a better discrimination ability.

3) Analysis of Computational Complexity: In this section, we take LPP, ELPP, regularization LPP (RLPP), FLPP, DLPP, EDLPP, and FDLPP methods as examples to discuss the computational complexity.

The execution of these methods is divided into four steps.

1) To construct the adjacency graph.
2) To compute the matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$, where the forms of the matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ for different methods are presented in Tables I and II.
3) To process the samples or the matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$. For the PCA + LPP method, this step is to reduce the dimension of the sample with the PCA method. For the ELPP method, this step is to compute the matrix exponential of the matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$. There are many methods for computing a matrix exponential [53]. The most widely used method is the scaling and squaring method [54]. For the FLPP method, this step is to compute the matrix functions $f(\boldsymbol{S}_1) = \boldsymbol{I} + \text{artanh}(\boldsymbol{S}_1)$ and $f(\boldsymbol{S}_2) = r\boldsymbol{I} + \boldsymbol{S}_2$. The major expense is to compute $\text{artanh}(\boldsymbol{S}_1)$. An efficient approximation method is introduced to compute $\text{artanh}(\boldsymbol{S}_1)$ in [55], and it is adopted to speed up the processing of $\text{artanh}(\boldsymbol{S}_1)$ from the computational point of view in this article.
4) To solve the generalized eigenvector problem. The total time complexity is evaluated in Table III.

As can be seen from Table III, the time complexity of FLPP is related to the power parameter s of the matrix $\boldsymbol{S}_1$. In real life, this value will not be too large. So the time complexity of FLPP is obviously lower than that of ELPP. It is comparable to that of PCA + LPP and RLPP in real life. The time complexity of the FDLPP method can be similarly compared.

VI. Experiments

To evaluate the proposed new matrix transformation framework, we make the experiments on the handwritten digit

TABLE III
ANALYSIS OF COMPUTATIONAL COMPLEXITY

Method	Total (Some items with a small proportion are deleted)
PCA+LPP	$\frac{13}{2}r^3+2mn^2+\left(\frac{3}{2}n+\log m\right)m^2+r^2n$
RLPP	$\frac{9}{2}n^3+2mn^2+\left(\frac{3}{2}m^2+3m\right)n$
ELPP	$\left(\frac{27}{2}+2C\right)n^3+2mn^2+\left(\frac{5}{2}m^2+3m\right)n$
FLPP	$\frac{9}{2}n^3+2mn^2+\left(\frac{3}{2}m^2+3m\right)n+2\sum_{s=1}^{k}\log s$
PCA+DLPP	$\frac{13}{2}r^3+2mn^2+\left(\frac{5}{2}n+\log m\right)m^2+r^2n$
EDLPP	$\left(\frac{27}{2}+2C\right)n^3+2mn^2+\left(\frac{5}{2}m^2+2m\right)n$
FDLPP	$\frac{9}{2}n^3+2mn^2+\left(\frac{5}{2}m^2+2m\right)n+2\sum_{s=1}^{k}\log s$

n: the number of samples,
m: the dimensionality of the sample,
r: the reduced dimensionality after PCA step,
C: $\pi_m+\left\lceil \log_2\left(\|A\|_1/\theta_m\right)\right\rceil$, where π_m,θ_m are the related parameters, $\|\bullet\|_1$ represents the 1-norm of a matrix, referred to [54]
s: the power of the matrix $\boldsymbol{S}_1$, referred to [55]

database, handwritten letter database, Georgia Tech, and FERET face database.

A. Experimental Datasets and Compared Methods

The handwritten databases are developed and published by New York University.[1] The Handwritten digit database consists of the numbers "0–9," each with 39 samples, each sample being a handwritten digital image. The Handwritten letter database consists of the 26 letters "A–Z," each with 39 samples, each sample being a handwritten letter image.

The Georgia Tech face database[2] contains photographs of 50 individuals taken during two or three meetings between January and November 2007. The image includes front and slanted faces under variations in facial expressions, illumination and proportions, and a cluttered background. Each image is aligned and scaled to 32 × 32 pixels in the experiments.

The FERET face database [56] is created by the FERET program at NIST in the United States. A subset of this face database was used in the experiment. The subset consists of 200 individuals, seven samples per person, for a total of 1400 face images. Each person is with different facial expressions, illumination, and pose. In the experiment, each image is aligned and scaled to 32 × 32 pixels.

Some sample images of the handwritten digital, letter, and two face image databases are shown in Fig. 4.

With the general matrix function dimensionality reduction framework, many new methods can be derived. Because LPP has been paid much attention by researchers, so we select FLPP and FDLPP to make experiments and compare them with the common methods to solve the SSS problem. In unsupervised cases, FLPP is compared with PCA+LPP, RLPP, LPPImprove3, and ELPP. FDLPP is compared with PCA+DLPP, RGDLPP, DLPP/MMC, and EDLPP. Besides, two classical methods, PCA and PCA+LDA [57] are also used to make experiments and compare.

[1]https://cs.nyu.edu/~roweis/data.html
[2]http://www.anefian.com/research/face_reco.htm

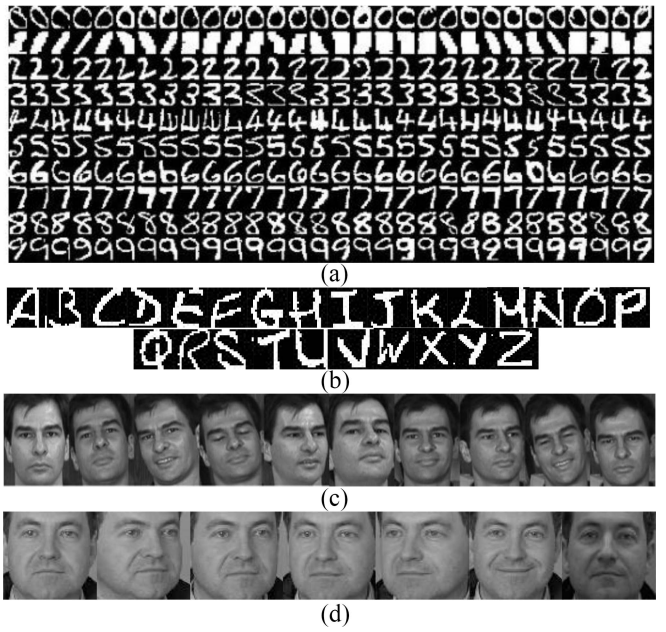

Fig. 4. Some sample images in the experiments. (a) Handwritten digit database. (b) Handwritten letter database. (c) Georgia Tech database. (d) FERET database.

B. Experimental Setup and Results

For the PCA+LPP, PCA + DLPP, and LDA methods, there is a stage of using the PCA technique to reduce the dimensionality of the original sample. For all the datasets, 99% of the principal components are reserved in the PCA step.

In each dataset, p images are randomly chosen to form a training set, and the remaining images are used as a test image set. The p value was set as different numbers for the different datasets. That is, $p = 3, 5, 7, p = 3, 5, 7, p = 7, 8, 9$, and $p = 3, 4, 5$ for Handwritten digit, Handwritten letter, Georgia Tech, and FERET databases, respectively.

For each method, for the given p, the reduced dimensionality is incremented from 10 to 100 at the step size 5. For the given p and subspace dimension, the neighborhood size parameter k is taken from and with step size = 10, where N is the training sample number. So there are many recognition rates are corresponding to the different k. The best recognition rate corresponding to the best k value is used as the recognition rate in the current p and subspace dimensions. Then, for each of the subspace dimension, we may obtain the corresponding recognition rate. In the experiments, we calculate ten times for the given p (at each time, p images are randomly taken as training samples, and the subspace dimension is incremented from 10 to 100 at the step size 5). Thus, there are ten recognition rates for each subspace dimension, and then their average value is regarded as the result in the current p and subspace dimension. The best recognition rate from the best subspace dimension is regarded as the result of the training sample = p.

TABLE IV
BEST AVERAGE RECOGNITION ACCURACY (IN PERCENT), STANDARD DEVIATION, AND THE OPTIMAL DIMENSION (IN PARENTHESES) ON THE HANDWRITTEN DIGIT SET OVER TEN RANDOM SPLITS

Method	3 trains	5 trains	7 trains
PCA	58.06±5.61(15)	67.38±2.74(15)	71.72±1.96(25)
PCA+LPP	53.45±3.97(27)	60.36±2.85(10)	65.97±2.99(10)
RLPP	55.75±5.14(35)	64.18±5.06(10)	68.88±1.15(10)
LPPImprove3	60.42±4.16(100)	72.03±3.35(20)	78.85±2.01(25)
ELPP	57.06±4.24(100)	68.50±3.08(10)	74.09±3.07(15)
FLPP	**65.97±4.20(45)**	**74.79±2.56(15)**	**80.41±2.04(80)**
LDA	64.00±2.84(9)	70.17±3.96(9)	73.38±2.37(9)
PCA+DLPP	61.48±3.38(15)	73.24±4.18(20)	77.81±4.72(15)
DLPP/MMC	72.04±6.40(40)	76.37±4.73(40)	79.58±4.55(100)
RGDLPP	71.54±5.14(40)	76.83±4.25(50)	81.67±4.76(85)
EDLPP	72.75±6.98(40)	78.43±3.84(20)	83.54±3.26(90)
FDLPP	**74.35±6.33(40)**	**80.59±3.57(70)**	**85.85±2.86(90)**

TABLE V
BEST AVERAGE RECOGNITION ACCURACY (IN PERCENT), STANDARD DEVIATION, AND THE OPTIMAL DIMENSION (IN PARENTHESES) ON THE HANDWRITTEN LETTERS SET OVER TEN RANDOM SPLITS

Method	3 trains	5 trains	7 trains
PCA	52.78±1.77(35)	55.79±1.42(30)	58.12±0.86(25)
PCA+LPP	42.47±3.12(70)	49.77±1.46(15)	53.80±1.53(20)
RLPP	46.49±3.02(70)	54.98±1.82(90)	60.65±1.90(20)
LPPImprove3	49.31±2.78(60)	59.71±1.56(80)	63.40±2.64(25)
ELPP	45.88±2.14(60)	56.08±1.69(100)	65.95±1.95(25)
FLPP	**53.56±2.64(60)**	**62.05±3.00(40)**	**67.47±1.72(25)**
LDA	56.00±2.84(25)	58.17±3.96(25)	60.22±2.37(25)
PCA+DLPP	57.56±1.63(55)	59.26±4.75(35)	61.65±0.77(50)
DLPP/MMC	51.47±4.16(40)	53.51±2.51(40)	55.30±1.08(75)
RGDLPP	59.90±2.33(95)	61.78±1.78(55)	63.90±2.08(15)
EDLPP	54.98±2.01(75)	63.46±1.69(60)	64.94±0.89(85)
FDLPP	**57.41±0.59(75)**	**65.38±2.33(60)**	**70.27±0.07(60)**

TABLE VI
BEST AVERAGE RECOGNITION ACCURACY (IN PERCENT), STANDARD DEVIATION, AND THE OPTIMAL DIMENSION (IN PARENTHESES) ON THE GEORGIA TECH FACE SET OVER TEN RANDOM SPLITS

Method	7 trains	8 trains	9 trains
PCA	68.60±1.55(65)	70.97±2.97(50)	72.84±0.98(95)
PCA+LPP	54.16±1.28(50)	56.19±0.72(45)	56.22±3.00(45)
RLPP	69.00±2.12(75)	71.56±2.75(45)	73.00±2.81(50)
LPPImprove3	70.65±2.58(70)	72.17±1.86(85)	73.73±1.93(60)
ELPP	70.17±0.72(45)	72.47±1.74(45)	74.00±2.33(50)
FLPP	**72.50±2.17(55)**	**73.52±1.41(70)**	**76.22±3.01(80)**
LDA	72.50±1.50(49)	73.86±2.87(49)	75.55±3.88(49)
PCA+DLPP	77.00±1.50(20)	81.24±2.59(20)	81.56±1.34(25)
DLPP/MMC	69.5±0.95(85)	68.19±3.21(55)	66.11±0.51(80)
RGDLPP	79.65±1.98(75)	80.78±2.67(80)	82.22±2.43(60)
EDLPP	81.83±1.25(80)	84.19±2.14(55)	84.22±2.59(60)
FDLPP	**83.08±1.15(85)**	**86.19±2.10(60)**	**86.33±1.21(50)**

For each method and each training sample p, the best recognition rates, standard deviation, and the optimal subspace dimension are reported in Tables IV–VII.

Generally, the recognition performances of these methods are also dependent on the reduced subspace dimension. Based on the above experiments, we collate the recognition results versus the variations of dimensions and display them in Figs. 5–8. In the figures, the dotted lines indicate the unsupervised methods and the solid lines indicate the supervised methods, respectively.

TABLE VII
BEST AVERAGE RECOGNITION ACCURACY (IN PERCENT), STANDARD DEVIATION, AND THE OPTIMAL DIMENSION (IN PARENTHESES) ON THE FERET FACE SET OVER TEN RANDOM SPLITS

Method	3 trains	4 trains	5 trains
PCA	58.82±6.92(100)	65.87±8.84(70)	68.20±12.68(95)
PCA+LPP	49.90±5.47(100)	51.90±6.71(100)	55.80±5.18(100)
RLPP	65.00±7.18(85)	69.16±6.78(65)	72.00±0.50(70)
LPPImprove3	51.67±6.63(70)	55.88±5.64(55)	58.67±5.63(80)
ELPP	60.82±12.47(90)	67.50±10.71(95)	70.95±6.10(100)
FLPP	**70.35±7.47(80)**	**72.17±6.08(90)**	**74.33±1.52(70)**
LDA	67.30±6.56(100)	75.30±7.81(35)	81.00±7.19(35)
PCA+DLPP	71.17±6.54(40)	82.22±2.11(75)	82.27±0.80(45)
RGDLPP	73.17±5.81(100)	79.93±7.93(90)	82.73±7.77(60)
DLPP/MMC	71.29±4.98(65)	69.00±6.78(75)	67.75±13.85(45)
EDLPP	76.54±5.09(40)	80.22±5.19(75)	86.25±8.87(30)
FDLPP	**80.17±4.52(40)**	**83.94±5.96(75)**	**89.67±0.14(45)**

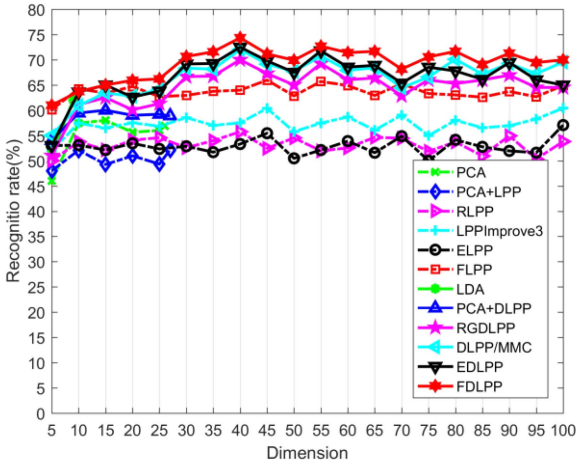

Fig. 5. Recognition rates versus the subspace dimension on the handwritten digit database (three training samples).

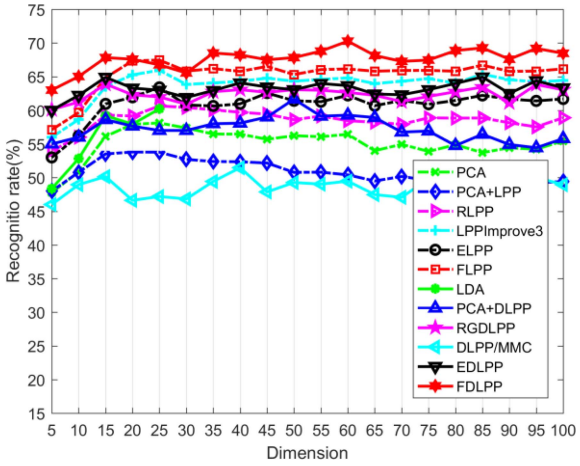

Fig. 6. Recognition rates versus the subspace dimension on the handwritten letters database (seven training samples).

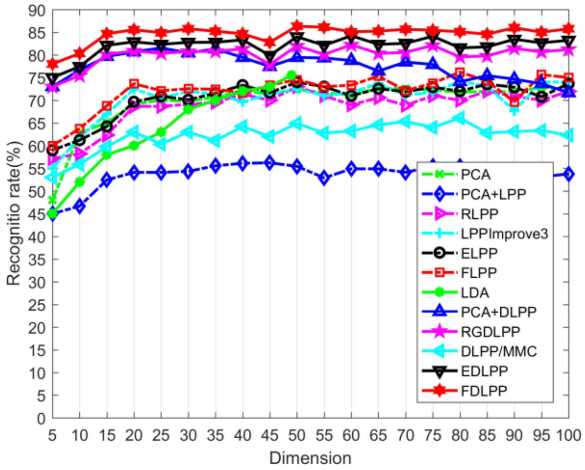

Fig. 7. Recognition rates versus the subspace dimension on the Georgia Tech face database (nine training samples).

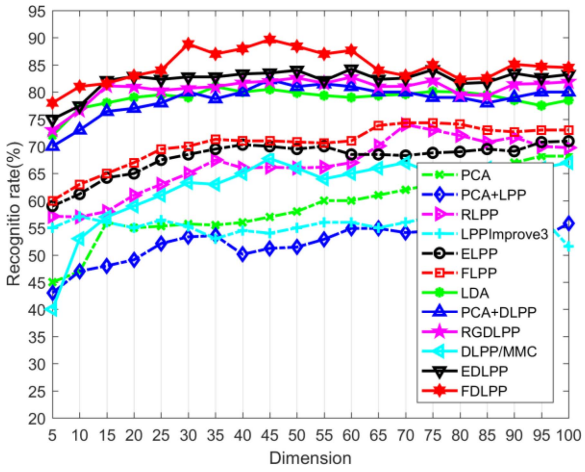

Fig. 8. Recognition rates versus the subspace dimension on the FERET face database (five training samples).

C. Analysis of Neighborhood Size Parameter k

The recognition results are also related to the neighborhood size parameter k. To evaluate the sensitivity of the recognition results versus the parameter k, we make an experiment for the FLPP method. The recognition results are shown in Fig. 9.

In the experiment, we randomly take the training sample number p from the above four databases, take the neighborhood size parameter k from 2 to 50 at step size $= 2$, and set the subspace dimension to 60. For the other parameter setup, similar results can be obtained. As can be seen from Fig. 9, the recognition rates of the FLPP methods show stable and robust when the parameter k varies. It indicates that the FLPP method is not sensitive to the parameter k. Although the k value is only taken from 2 to 50 in this experiment, in fact, it is no much sense to take a large k value in real life. As in [43] and [58], the value of k is only taken in a small range.

Due to the design of the DLPP method, it is not related to the neighborhood size parameter k, so the analysis of DLPP is not involved in the experiment.

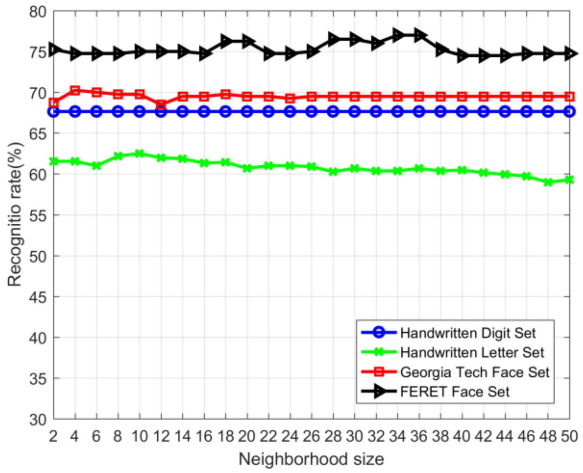

Fig. 9. Recognition results of the FLPP method versus the parameter k .

TABLE VIII
COMPARISON OF COMPUTATIONAL TIME (S) ON DIFFERENT DATABASES

Dataset	Handwritten letter (5 trains)	Georgia Tech (7 trains)	FERET (5 trains)
PCA+LPP	1.04	1.01	2.33
ELPP	1.43	5.08	6.41
FLPP	**1.08**	**1.37**	**2.92**
PCA+DLPP	0.75	1.39	4.43
EDLPP	2.53	5.18	5.92
FDLPP	**0.72**	**1.19**	**2.32**

D. Comparison of Computational Time

In the section, the time complexity of PCA+LPP, ELPP, FLPP, PCA+DLPP, EDLPP, and FDLPP is compared on Handwritten letters, Georgia Tech, and FERET face databases. In the experiment, the number of training samples is randomly set to $p = 5, 7$, and 5 for Handwritten letters, Georgia Tech, and FERET databases, respectively. The subspace dimension of dimensionality reduction is set to 40, and the neighborhood size parameter k is set to 12. Under this parameter setting, each method is calculated ten times, and the average value is taken as the computational time of this method. The results are reported in Table VIII. Because the time complexity of FLPP is comparable to that of RLPP and LPPImprove3, the time complexity of FDLPP is comparable to that of RGDLPP and DLPP/MMC, and the computational time of RLPP, LPPImprove3, RGDLPP, and DLPP/MMC are not listed. It can be seen from Table VIII that the computational time of the proposed matrix function transformation methods is comparable to or lower than that of PCA + LPP and PCA+DLPP, and obviously lower than that of the matrix exponential methods.

E. Analysis

From the experiment results, it can be observed as follows.

1) Compared with the other methods, the FDLPP method has better performance among all subspace dimensions in all databases.
2) Among the unsupervised methods, the proposed FLPP method shows better performance than the other

methods in all databases. Generally, the performances of unsupervised methods are lower than that of the supervised methods, because the class label information is used in supervised learning. But, in the Handwritten letter database, the recognition rate of the FLPP method exceeds that of the supervised methods except for FDLPP. In other databases, the recognition rate of the FLPP method may exceed that of some of the supervised methods.

3) The DLPP/MMC method only has a high recognition rate on the Handwritten digit database and is even lower than the unsupervised method on other databases. It may be due to the parameters set in the DLPP/MMC method.
4) The FLPP method is not sensitive to the neighborhood size parameter k. It is a good performance.
5) The time complexity of the FLPP method is comparable to that of the PCA + LPP, RLPP, and LPImprove3 methods, but the performance of the FLPP method is better than theirs. Compared with the ELPP method, the FLPP method has lower time complexity and better performance. In the supervised methods, the comparison of the FDLPP method with other supervised methods has similar results.

VII. Conclusion

Many dimensionality reduction methods in manifold learning have a common problem, that is, the SSS problem. Starting from solving the SSS problem, this article first established a unified criterion function form of these methods and then proposed a general matrix function dimensionality reduction framework. The framework may be used as a general platform, from which a series of new dimensionality reduction methods can be derived. In the view of solving the SSS problem and improving the ability of pattern classification, this article discussed how to choose suitable functions to construct the framework. As an application, we have proposed a new matrix function dimensionality reduction framework. Compared with existing methods for solving the SSS problem, these methods have three advantages: 1) they have better pattern classification performance; 2) they have lower computational complexity; and 3) they are not sensitive to the neighborhood size parameter k.

The further work of the research is to find some better functions to construct a matrix function transformation under this general matrix transformation framework so that the dimensionality reduction methods can obtain better performance. Also, this article only discussed the dimensionality reduction methods in the unsupervised and supervised cases. This framework may be used in the semisupervised learning case.

References

[1] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cogn. Neurosci.*, vol. 3, no. 1, pp. 71–86, Jan. 1991.

[2] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Ann. Eugen.*, vol. 7, no. 2, pp. 179–188, 1936.

[3] C. R. Rao, "The utilization of multiple measurements in problems of biological classification," *J. Roy. Stat. Soc. B (Methodol.)*, vol. 10, no. 2, pp. 159–203, 1948.

[4] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Proc. Conf. Adv. NIPS*, vol. 14, 2002, pp. 585–591.

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

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

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

[8] C. Hou, F. Nie, D. Yi, and D. Tao, "Discriminative embedded clustering: A framework for grouping high-dimensional data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 3, pp. 1287–1299, Jun. 2015.

[9] W. Wu, S. Kwong, J. Hou, Y. Jia, and H. H. S. Ip, "Simultaneous dimensionality reduction and classification via dual embedding regularized nonnegative matrix factorization," *IEEE Trans. Image Process.*, vol. 28, no. 8, pp. 3836–3847, Aug. 2019, doi: 10.1109/TIP.2019.2907054.

[10] A. L. M. Levada, "A review on unsupervised learning for dimensionality reduction: From linear methods to manifold learning," unpublished, doi: 10.13140/RG.2.2.17736.37129/1.

[11] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimension reduction via local tangent space alignment," *SIAM J. Sci. Comput.*, vol. 26, no. 1, pp. 313–338, 2004.

[12] J. Yu, M. Tan, H. Y. Zhang, Y. Rui, and D. C. Tao, "Hierarchical deep click feature prediction for fine-grained image recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Jul. 30, 2019, doi: 10.1109/TPAMI.2019.2932058.

[13] L. Zhang, L. Zhang, B. Du, J. You, and D. Tao, "Hyperspectral image unsupervised classification by robust manifold matrix factorization," *Inf. Sci.*, vol. 485, pp. 154–169, Jun. 2019.

[14] L. Zhang, Q. Zhang, B. Du, D. Tao, and J. You, "Robust manifold matrix factorization for joint clustering and feature extraction," in *Proc. 31st AAAI Conf.*, San Francisco, CA, USA, 2017, pp. 1662–1668.

[15] L. Zhang, Q. Zhang, L. Zhang, D. Tao, X. Huang, and B. Du, "Ensemble manifold regularized sparse low-rank approximation for multiview feature embedding," *Pattern Recognit.*, vol. 48, no. 10, pp. 3102–3112, 2015.

[16] J. Yu, X. K. Yang, F. Gao, and D. C. Tao, "Deep multimodal distance metric learning using click constraints for image ranking," *IEEE Trans. Cybern.*, vol. 47, no. 22, pp. 4014–4024, Dec. 2017.

[17] J. Yu, D. C. Tao, M. Wang, and Y. Rui, "Learning to rank using user clicks and visual features for image retrieval," *IEEE Trans. Cybern.*, vol. 45, no. 4, pp. 767–779, Apr. 2015.

[18] J. Yu, C. Y. Zhu, J. Zhang, Q. M. Huang, and D. C. Tao, "Spatial pyramid-enhanced NetVLAD with and weighted triplet loss for place recognition," *IEEE Trans. Neural Netw. Learn. Syst*, vol. 31, no. 2, pp. 661–674, Feb. 2020, doi: 10.1109/TNNLS.2019.2908982.

[19] X. C. Wang, Z. Li, L. Zhang, and J. S. Yuan, "Grassmann hashing for approximate nearest neighbor search in high dimensional space," in *Proc. IEEE Conf. Multimedia Expo ICME*, Barcelona, Spain, 2011, pp. 1–6.

[20] X. C. Wang, Z. Li, and D. C. Tao, "Subspaces indexing model on Grassmann manifold for image search," *IEEE Trans. Image Process.*, vol. 20, no. 9, pp. 2627–2635, Sep. 2011.

[21] X. C. Wang, W. Bian, and D. C. Tao, "Grassmannian regularized structured multi-view embedding for image classification," *IEEE Trans. Image Process.*, vol. 22, no. 7, pp. 2646–2660, Jul. 2013.

[22] X. F. He and P. Niyogi, "Locality preserving projections," in *Proc. Adv. NIPS*, vol. 16, 2004, pp. 153–160.

[23] D. Cai, X. F. He, and J. W. Han, "Isometric projection," in *Proc. AAAI*, Vancouver, BC, Canada, 2007, pp. 528–533.

[24] X. F. He, D. Cai, S. Yan, and H. Zhang, "Neighborhood preserving embedding," in *Proc. Conf. Comput. Vis. (ICCV)*, Beijing, China, 2005, pp. 1208–1213.

[25] 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.

[26] Y. Pang, N. Yu, H. Li, R. Zhang, and Z. Liu, "Face recognition using neighborhood preserving projections," in *Proc. Pac. Rim Conf. Multimedia (PCM)*, 2005, pp. 854–864.

[27] T. Zhang, J. Yang, D. Zhao, and X. Ge, "Linear local tangent space alignment and application to face recognition," *Neurocomputing*, vol. 70, pp. 1547–1553, Mar. 2007.

[28] H. Chen, H. Chang, and T. Liu, "Local discriminant embedding and its variants," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, San Diego, CA, USA, 2005, pp. 846–853.
[29] W. Yu, X. Teng, and C. Liu, "Face recognition using discriminant locality preserving projections," *Image Vis. Comput.*, vol. 24, no. 3, pp. 239–248, Mar. 2006.
[30] H. Zhao, S. Sun, Z. Jing, and J.-Y. Yang, "Local structure based supervised feature extraction," *Pattern Recognit.*, vol. 39, no. 8, pp. 1546–1550, Aug. 2006.
[31] P. Y. Han, A. T. B. Jin, and F. S. Abas, "Neighbourhood preserving discriminant embedding in face recognition," *J. Vis. Commun. Image Represent.*, vol. 9, no. 8, pp. 532–542, 2009.
[32] S. Yan, D. Xu, B. Zhang, H. J. 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.
[33] D. Xu, S. Yan, D. Tao, S. Lin, and H.-J. Zhang, "Marginal Fisher analysis and its variants for human gait recognition and content-based imageretrieval," *IEEE Trans. Image Process.*, vol. 16, no. 11, pp. 2811–2821, Nov. 2007.
[34] M. H. Yang, "Discriminant isometric mapping for face recognition," in *Proc. Int. Conf. Comput. Vis. Syst. (ICVS)*, 2003, pp. 470–480.
[35] A. Younesi and M. C. Amirani, "Palmprint identification using isometric projection and linear discriminant analysis," in *Proc. 1st Conf. Commun. Signal Process. Appl. (ICCSPA)*, 2013, pp. 510–517.
[36] Y. Xu, A. Zhong, J. Yang, and D. Zhang, "LPP solution schemes for use with face recognition," *Pattern Recognit.*, vol. 43, no. 12, pp. 4165–4176, 2010.
[37] D. Cai, X. He, and J. Han, "Spectral regression for efficient regularized subspace learning," in *Proc. IEEE Conf. Comput. Vis. (ICCV)*, Rio de Janeiro, Brazil, 2007, pp. 1–8.
[38] G. F. Lu, Z. Lin, and Z. Jin, "Face recognition using regularised generalized discriminant locality preserving projections," *IET Comput. Vis.*, vol. 5, no. 2, pp. 107–116, 2011.
[39] X. Chen and J. Zhang, "A novel maximum margin neighborhood preserving embedding for face recognition," *Future Gener. Comput. Syst.*, vol. 28, no. 1, pp. 212–217, 2012.
[40] G. F. Lu, Z. Lin, and Z. Jin, "Face recognition using discriminant locality preserving projections based on maximum margin criterion," *Pattern Recognit.*, vol. 43, no. 10, pp. 3572–3579, 2010.
[41] R. S. Ran, B. Fang, and X. G. Wu, "Exponential neighborhood preserving embedding for face recognition," *IEICE Trans. Inf. Syst.*, vol. E101-D, no. 5, pp. 1410–1420, May 2018.
[42] S. Wang, H.-L. Chen, X.-J. Peng, and C. Zhou, "Exponential locality preserving projections for small sample size problem," *Neurocomputing*, vol. 74, no. 17, pp. 3654–3662, 2001.
[43] S. C. Huang and L. Zhuang, "Exponential discriminant locality preserving projection for face recognition," *Neurocomputing*, vol. 208, pp. 373–377, Oct. 2016.
[44] G. F. Lu, Y. Wang, and J. Zou, "Matrix exponential based discriminant locality preserving projections for feature extraction," *Neural Netw.*, vol. 97, pp. 127–136, Jan. 2018.
[45] F. Dornaika and A. Bosaghzadeh, "Exponential local discriminant embedding and its application to face recognition," *IEEE Trans. Cybern.*, vol. 43, no. 3, pp. 921–934, Jun. 2013.
[46] F. Dornaika and Y. E. Traboulsi, "Matrix exponential based semi-supervised discriminant embedding for image classification," *Pattern Recognit.*, vol. 61, pp. 92–103, Jan. 2017.
[47] R. S. Ran, B. Fang, and S. G. Zhang, "Face recognition based on exponential neighborhood preserving discriminant embedding," *J. Comput.*, vol. 30, no. 6, pp. 1–14, Dec. 2019.
[48] S. Wang, S. Yan, J. Yang, C. Zhou, and X. Fu, "A general exponential framework for dimensionality reduction," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 920–930, Feb. 2014.
[49] D. Cai, X. He, and J. Han, "Spectral regression: A unified approach for sparse subspace learning," in *Proc. IEEE Conf. Data Min. (ICDM)*, Omaha, NE, USA, 2007, pp. 73–82.
[50] D. Cai, X. He, and J. Han, "Spectral regression: A unified subspace learning framework for content-based image retrieval," in *Proc. Conf. Multimedia (ACM Multimedia)*, Augsburg, Germany, 2007, pp. 403–412.
[51] Z. Lai, Y. Xu, J. Yang, L. Shen, and D. Zhang, "Rotational invariant dimensionality reduction algorithms," *IEEE Trans. Cybern.*, vol. 47, no. 11, pp. 3733–3746, Nov. 2017.
[52] G. H. Golub and C. F. V. Loan, *Matrix Computation*, 4th ed. Baltimore, MD, USA: Johns Hopkins Univ. Press, 2013, pp. 513–515.
[53] C. Moler and C. V. Loan, "Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later," *SIAM Rev.*, vol. 45, no. 1, pp. 3–49, Feb. 2003.
[54] N. Higham, "The scaling and squaring method for the matrix exponential revisited," *SIAM J. Matrix Anal. Appl.*, vol. 26, no. 4, pp. 1179–1196, Jun. 2005.
[55] L. Dieci, "Considerations on computing real logarithms of matrices, hamiltonian logarithms, and skew-symmetric logarithms," *Linear Algebra Appl.*, vol. 244, pp. 35–54, Oct. 1996.
[56] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERETevaluation methodology for face-recognition algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1090–1104, Oct. 2000.
[57] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.
[58] J. Lu and Y. Tan, "Regularized locality preserving projections and its extensions for face recognition," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 3, pp. 963–985, Jun. 2010.

Ruisheng Ran received the B.S. degree in mathematics from Chongqing Normal University, Chongqing, China, in 1998, the M.S. degree in computational mathematics and the Ph.D. degree in computer application technology from the University of Electronic Science and Technology of China, Chengdu, China, in 2004 and 2006, respectively.

He is currently a Professor with the College of Computer and Information Science, Chongqing Normal University. He is doing pattern recognition and computer vision research.

Ji Feng received the Ph.D. degree in computer science from Chongqing University, Chongqing, China, in 2016.

He is currently a Lecturer with the Computer and Information Science College, Chongqing Normal University, Chongqing. His research interests include artificial intelligence and pattern recognition and applications.

Shougui Zhang received the B.S. degree from Chongqing Normal University, Chongqing, China, in 1998, and the M.S. and Ph.D. degrees from Chongqing University, Chongqing, in 2005 and 2012, respectively.

He is currently an Associate Professor with the College of Mathematical Sciences, Chongqing Normal University. His research interests include numerical solution of PDEs, and boundary element methods. He is particularly interested in computational methods and optimization application in variational inequalities.

Bin Fang (Senior Member, IEEE) received the B.S. degree in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 1989, the M.S. degree in electrical engineering from Sichuan University, Chengdu, China, in 1994, and the Ph.D. degree in electrical engineering from the University of Hong Kong, Hong Kong, in 2001.

He is currently a Professor with the Department of Computer Science, Chongqing University, Chongqing, China. He has published more than 100 technical papers. His research interests include computer vision, pattern recognition, medical image processing, biometrics applications, and document analysis.

Prof. Fang is an Associate Editor of the *International Journal of Pattern Recognition* and *Artificial Intelligence*.