International Journal of Machine Learning and Cybernetics (2022) 13:2013–2032
https://doi.org/10.1007/s13042-021-01502-6

ORIGINAL ARTICLE

Global structure-guided neighborhood preserving embedding for dimensionality reduction

Can Gao[1,2,3] · Yong Li[1,2,3] · Jie Zhou[1,2,3] · Witold Pedrycz[4,5] · Zhihui Lai[1,2,3] · Jun Wan[1,2,3] · Jianglin Lu[1,2,3]

Received: 24 December 2020 / Accepted: 29 December 2021 / Published online: 11 January 2022

Abstract

Graph embedding is one of the most efficient dimensionality reduction methods in machine learning and pattern recognition. Many local or global graph embedding methods have been proposed and impressive results have been achieved. However, little attention has been paid to the methods that integrate both local and global structural information without constructing complex graphs. In this paper, we propose a simple and effective global structure guided neighborhood preserving embedding method for dimensionality reduction called GSGNPE. Specifically, instead of constructing global graph, principal component analysis (PCA) projection matrix is first introduced to extract the global structural information of the original data, and then the induced global information is integrated with local neighborhood preserving structure to generate a discriminant projection. Moreover, the $L_{2,1}$-norm regularization is employed in our method to enhance the robustness to occlusion. Finally, we propose an iterative optimization algorithm to solve the proposed problem, and its convergence is also theoretically analyzed. Extensive experiments on four face and six non-face benchmark data sets demonstrate the competitive performance of our proposed method in comparison with the state-of-the-art methods.

Keywords Dimensionality reduction · Neighborhood preserving embedding · Global structure · Principal component analysis · Structured sparsity

1 Introduction

In the era of big data, the tasks to deal with, such as image classification, text analysis and gene selection, contain hundred and thousand of features, pose some substantial challenges for effective and efficient computing and analysis [1–7]. Dimensionality reduction can be considered as the process of removing irrelevant or redundant information from high-dimensional data [8, 9]. Since reducing the learning process, improving interpretability, and alleviating the problem of over-fitting, dimensionality reduction has become an important pre-processing step in machine learning, pattern recognition, and computer vision [10–12].

In recent decades, many linear dimensionality reduction methods [13–16] have been proposed. Among these methods, principal component analysis (PCA) [13] and linear discriminant analysis (LDA) [14] are the classic ones and widely used. PCA is an unsupervised dimensionality reduction method, which aims to find a projection that maximizes the overall variance [17]. Unlike PCA, LDA is a supervised method, which tries to find a projection that not only makes the samples within the same classes compact and also keeps the samples from different classes away from each other. In practical applications, however, the data does not always have an ideal linear structure [18, 19]. In these cases, PCA and LDA may not achieve the expected performance since they do not take into account the underlying nonlinear structure of the original data. To tackle this problem, many

✉ Jie Zhou
jie_jpu@163.com

1 College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, People's Republic of China

2 Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen 518060, Guangdong, People's Republic of China

3 SZU Branch, Shenzhen Institute of Artificial Intelligence and Robotics for Society, Shenzhen 518060, People's Republic of China

4 Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Canada

5 Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland

kernelized extensions of these linear dimensionality reduction methods [20–22] have been proposed. Nevertheless, due to the difficulty in determining kernel function, these methods are limited in practical applications.

The high-dimensional data is always in a low-dimensional manifold embedded in the original high-dimensional space [23]. Thus, to introduce the low-dimensional manifold of high-dimensional data into the low-dimensional space, many nonlinear manifold learning methods have been proposed, including locally linear embedding (LLE) [24], isometric mapping (ISOMAP) [23], and Laplacian eigenmap (LE) [25]. Though these manifold learning methods can well preserve the inherent geometric structure of the original data in the low-dimensional space [26], they directly obtain the low-dimensional embedding without the explicit mapping functions resulting in the "out-of-sample" problem. This inconvenience limits the applications of these methods in practice. The key to solving this problem is to obtain an explicit mapping function, which can easily obtain the low-dimensional embeddings of new samples. In the past few decades, many linear manifold learning methods have emerged, including locality preserving projections (LPP) [27], neighborhood preserving embedding (NPE) [28], neighborhood preserving projections (NPP) [29], sparsity preserving projections (SPP) [30]. Many of these methods are extensions of the existing nonlinear manifold learning methods. For example, LPP is a linear extension of LE and uses an affinity graph to make the projection data retain neighborhood structure; NPE is a linear extension of LLE and uses weighted graph to minimize the reconstruction error for keeping local structure of the original data. These linear manifold learning methods can not only retain the local manifold structure of the original data but also solve the "out-of-sample" problem.

However, there are still some potential drawbacks in nonlinear and linear manifold learning methods mentioned above. First, most of these methods neglect the global structure of the original data which is also useful to identify the underlying structural information of the data [31]. Second, the local structure graph depends on the pair-wise Euclidean distances between samples, which is easily contaminated by noise and outlier [32]. To address these problems, many graph embedding based methods [33–36] have been proposed, which can hold the global and local structure of the original data in low-dimensional space. These methods preserve the structural information by constructing the complex structure graphs. For example, Shen et al. [35] constructed a global structure graph by calculating the distance between all samples, which is combined with the local graph embedding based method to get the low-dimensional embedding. In [36], Gou et al. assumed that the samples within the same classes will hold the similar sparse reconstructions. By constructing the global and local structure graphs, they gave the definition of sparsity and geometry preserved scattering and employed maximum margin criterion (MMC) [22] to obtain the low-dimensional embedding.

Although keeping the global and local structure simultaneously in the low-dimensional space can effectively improve the performance of dimensionality reduction, the construction of structure graphs in the existing methods are very complicated. Also, it is difficult to find an appropriate method to efficiently and easily combine the global and local structural information in practical applications. To deal with this problem, we propose a simple and effective dimensionality reduction method named GSGNPE (global structure guided neighborhood preserving embedding), which can simultaneously retain the global and local structure of the original data in low-dimensional space. In detail, we use a concise and efficient least-square term to obtain the global structure information from the PCA projection matrix which can retain the global Euclidean structure of the original data. On this basis, we modify the NPE loss function so that it can hold the global and local structure of the original data concurrently. In addition, the $L_{2,1}$-norm regularization is utilized in GSGNPE model to make the final projection matrix have the sparse structure, promote effective learning of global structure information, and reduce the interference of global structure information on local structure information, which helps to retain essential structural information and makes GSGNPE robustness to occlusion and noise. Hence, the performance of GSGNPE can be further improved. The main contributions of this work are summarized as follows:

1) A novel unsupervised graph embedding based dimensionality reduction method named GSGNPE is proposed. Instead of constructing the global structure graph, GSGNPE introduces the global structural information in the projection matrix obtained by PCA into the NPE to guide the retention of structural information, which provides a simple and effective way to extract the features with global and local structural information.
2) Consider the negative effect of noise and occlusion, GSGNPE integrates $L_{2,1}$-norm regularization into its objective function, which could effectively promote the retention of essential structural information and enhance the robustness to occlusion and noise. The experiments on the face data sets show that GSGNPE achieves better results under different sizes and positions of occlusion.
3) An iterative algorithm is elaborately designed for solving the resulting optimization problem of the proposed GSGNPE, and the corresponding convergence is analyzed and proved theoretically. Extensive experimental results demonstrate the superior performance and fast convergence of the proposed GSGNPE.

The rest of this paper is organized as follows. Section 2 presents some notations and definitions, and gives a brief

review of the related works. Section 3 describes the proposed GSGNPE method and the corresponding optimization algorithm, and gives the convergence analysis. A series of experimental results and analysis are given in Sect. 4. Finally, the conclusions are drawn in Sect. 5.

2 Related works

In this section, we first give some notation and definitions, and then briefly review some related methods, including PCA, LPP, and NPE.

2.1 Notation and definitions

Suppose that $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \Re^{d\times n}$ is the centralized data matrix, and $\mathbf{x}_i \in \Re^d$ is the ith sample of $\mathbf{X}$. Let x_{ij} be the element in the ith row and jth column of the matrix $\mathbf{X}$, and $\mathbf{x}^i$ is the ith row vector of data matrix $\mathbf{X}$. Then, the $L_{2,1}$ -norm of data matrix $\mathbf{X}$ is defined as follows:

$$||\mathbf{X}||_{2,1} = \sum_{i=1}^{d} \sqrt{\sum_{j=1}^{n} x_{ij}^2} = \sum_{i=1}^{d} ||\mathbf{x}^i||_2, \tag{1}$$

The goal of the linear dimensionality reduction methods is to learn a projection matrix $\mathbf{P} \in \Re^{d\times c}$ that can project a high-dimensional sample $\mathbf{x}_i$ to a low-dimensional one with $\mathbf{y}_i = \mathbf{P}^T\mathbf{x}_i$, where $c \ll d$.

2.2 PCA

PCA is one of the most commonly used dimensionality reduction methods in pattern recognition and machine learning. The basic idea of PCA is to maximize the overall variance of the samples, and the projection data preserves the global Euclidean structure of the original data [37]. The objective function of PCA is defined as:

$$\max_{\mathbf{\Theta}} \sum_{i=1}^{n} ||\mathbf{\Theta}^T\mathbf{x}_i||_2^2 = tr(\mathbf{\Theta}^T\mathbf{\Sigma}\mathbf{\Theta}), \quad \text{s.t. } \mathbf{\Theta}^T\mathbf{\Theta} = I, \tag{2}$$

where $\mathbf{\Theta} \in \Re^{d\times c}$ is the projection matrix, and $\mathbf{\Sigma} = \mathbf{X}\mathbf{X}^T$ is the covariance matrix of the samples. The optimal solution $\mathbf{\Theta}$ consists of the eigenvectors corresponding to the first c largest eigenvalues of the covariance matrix $\mathbf{\Sigma}$.

2.3 LPP

Unlike PCA, LPP is a local graph embedding-based method which aims to find a low-dimensional space retaining the local structure of the original data. The objective function is defined as follows:

$$\min_{\mathbf{P}} \sum_{ij} ||\mathbf{P}^T\mathbf{x}_i - \mathbf{P}^T\mathbf{x}_j||_2^2 w_{ij}, \tag{3}$$

where $\mathbf{P} \in \Re^{d\times c}$ is the projection matrix, and w_{ij} is an entry of $\mathbf{W} \in \Re^{n\times n}$ that describes the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$, defined as:

$$w_{ij} = \begin{cases} exp(-||\mathbf{x}_i - \mathbf{x}_j||^2/t), & \text{if nodes } i \text{ and } j \text{ are adjacent,} \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

With the constraint of $\mathbf{P}^T\mathbf{X}\mathbf{D}\mathbf{X}^T\mathbf{P} = \mathbf{I}_c$, where $\mathbf{I}_c$ is a $c \times c$ identity matrix, the objective function (3) can be written as:

$$\min_{\mathbf{P}} tr(\mathbf{P}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{P}), \quad \text{s.t. } \mathbf{P}^T\mathbf{X}\mathbf{D}\mathbf{X}^T\mathbf{P} = \mathbf{I}_c \tag{5}$$

where $\mathbf{D} \in \Re^{n\times n}$ is a diagonal matrix with $\mathbf{D}_{ii} = \mathbf{\Sigma}_j w_{ij}$, and $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is the graph Laplacian matrix. The objective function (5) can be converted to the following eigenvalue problem:

$$\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{P} = \mathbf{X}\mathbf{D}\mathbf{X}^T\mathbf{P}\mathbf{E}, \tag{6}$$

where $\mathbf{E}$ is the eigenvalue matrix. LPP could keep the neighboring high-dimensional samples adjacent to each other in the low-dimensional space.

2.4 NPE

NPE is a linear approximation extension of LLE. Unlike LPP, NPE finds a low-dimensional embedding by minimizing the local reconstruction error between the samples and their neighbors [30]. The calculation process of the weight matrix $\mathbf{W}$ in NPE is similar to that of LLE, which is defined as:

$$\min_{\mathbf{W}} \sum_{i}^{n} \left\| \mathbf{x}_i - \sum_{j} w_{ij}\mathbf{x}_j \right\|_2^2, \quad \text{s.t. } \sum_{j} w_{ij} = 1, \tag{7}$$

where $\mathbf{x}_j$ is one of the k neighbors of $\mathbf{x}_i$, and the KNN algorithm is often used to determine the neighbors of $\mathbf{x}_i$. The objective function of NPE is defined as follows:

$$\min_{\mathbf{P}} \sum_{i}^{n} \left\| \mathbf{P}^T\mathbf{x}_i - \sum_{j}^{k} w_{ij}\mathbf{P}^T\mathbf{x}_j \right\|_2^2, \quad \text{s.t.} \quad \mathbf{P}^T\mathbf{X}\mathbf{X}^T\mathbf{P} = \mathbf{I}_c, \tag{8}$$

where $\mathbf{P} \in \Re^{d\times c}$ is the projection matrix, and $\mathbf{I}_c$ is a $c \times c$ identity matrix. The objective function (8) can be written as:

$$\min_{\mathbf{P}} tr(\mathbf{P}^T\mathbf{X}\mathbf{M}\mathbf{X}^T\mathbf{P}), \quad \text{s.t.} \quad \mathbf{P}^T\mathbf{X}\mathbf{X}^T\mathbf{P} = \mathbf{I}_c, \tag{9}$$

where $\mathbf{M}$ is a symmetric matrix and $\mathbf{M} = (\mathbf{I} - \mathbf{W})^T(\mathbf{I} - \mathbf{W})$. The minimization problem (9) can be converted to the following eigenvalue problem:

$$\mathbf{XMX}^T\mathbf{P} = \mathbf{XX}^T\mathbf{PE}, \tag{10}$$

where $\mathbf{E}$ is the eigenvalue matrix.

3 The proposed method

In this section, we propose a simple and effective unsupervised graph embedding based method for dimensionality reduction, which can maintain the global and local structural information of the original data in low-dimensional space. We first introduce the motivation of our proposed method. After that we will detailedly describe the proposed method and optimization algorithm. Finally, the convergence analysis is given.

3.1 Motivation

Many graph embedding based subspace learning methods are implemented by finding the low-dimensional embedding that preserves the local structure of the original data. Among these methods, LLE and its extension methods get the low-dimensional manifold embedding by reconstructing the linear relationship between the samples and their neighborhoods in the low-dimensional space. Neighborhood relations in these methods represent the similarity of samples, which can gather together similar samples [38]. Most of these graph embedding based methods can be expressed by [35]:

$$\min_{\mathbf{p}_i} \mathbf{p}_i^T\mathbf{XGX}^T\mathbf{p}_i, \quad \text{s.t.} \quad \mathbf{p}_i^T\mathbf{XDX}^T\mathbf{p}_i = 1, \tag{11}$$

where $\mathbf{p}_i$ is the column vector of projection matrix $\mathbf{P} = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_c\} \in \mathfrak{R}^{d\times c}$, $\mathbf{G} \in \mathfrak{R}^{n\times n}$ is an affinity graph which represents the local neighborhood structure of samples, and $\mathbf{D}$ is the discriminate matrix. The difference between these graph embedding based methods lies in the way of the construction of affinity graph $\mathbf{G}$. However, these methods do not take into account the global structure of the original data, the obtained projection matrix $\mathbf{P}$ cannot retain the global structure in the low-dimensional space. When the overall structure of the data is relatively scattered, the local graph embedding based methods may not achieve good performance. Therefore, it is worthwhile to investigate the dimensionality reduction method preserving both the global and local structure of the original data in the low-dimensional space, simultaneously. By constructing a global graph, the global structural information of the original data could be captured. However, it is not easy to construct a suitable global structure graph for dimensionality reduction, and the effectiveness of the constructed graph may not be guaranteed. While some classic linear dimensionality reduction methods, such as PCA, can hold the global structure of the original data, and the effectiveness of these methods has been widely confirmed [39]. Inspired by this, we propose a simple and effective method to preserve the global and local graph structural information of the original data in the low-dimensional space, which avoids the construction of global structure graph and ensures the effectiveness.

3.2 The objective function of GSGNPE

Let $\mathbf{\Theta} = \{\theta_1, \theta_2, \ldots, \theta_c\} \in \mathfrak{R}^{d\times c}$ be the projection matrix of PCA, $\theta_i \in \mathfrak{R}^d$ is the column vector of $\mathbf{\Theta}$, and $\mathbf{A} \in \mathfrak{R}^{d\times d}$ is an orthogonal square matrix with $\mathbf{A}^T\mathbf{A} = \mathbf{AA}^T = \mathbf{I}$. Consider the following optimization problem:

$$\min_{\mathbf{P},\mathbf{A}} \sum_{i=1}^{c} \left\|\theta_i - \mathbf{A}^T\mathbf{p}_i\right\|_2^2, \quad \text{s.t.} \quad \mathbf{A}^T\mathbf{A} = \mathbf{I}, \tag{12}$$

where $\mathbf{P} = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_c\} \in \mathfrak{R}^{d\times c}$, $\mathbf{p}_i \in \mathfrak{R}^d$ is the column vector of $\mathbf{P}$. When $\mathbf{A}$ is fixed, we have:

$$\mathbf{p}_i = (\mathbf{AA}^T)^{-1}\mathbf{A}\theta_i = \mathbf{A}\theta_i. \tag{13}$$

Optimization problem (12) aims to project $\mathbf{p}_i$ into θ_i, which passes the information of θ_i to $\mathbf{p}_i$. And this conclusion can also be drawn from (13). The projection matrix $\mathbf{\Theta}$ obtained by PCA maximizes the overall variance, which also keeps the global Euclidean structure of the original data [37]. Naturally, in the process of dynamically fitting θ_i, $\mathbf{p}_i$ efficiently gains the global Euclidean structural information in θ_i. Inspired by (12), we modify the model of NPE so that it can retain the global and local structural information of the original data in the low-dimensional space. Take the following optimization problem into consideration:

$$\begin{aligned}\min_{\mathbf{P},\mathbf{A}} &\sum_{i}^{n} \left\|\mathbf{P}^T\mathbf{x}_i - \sum_j w_{ij}\mathbf{P}^T\mathbf{x}_j\right\|_F^2 + \alpha\sum_{i=1}^{c}||\theta_i - \mathbf{A}^T\mathbf{p}_i||_2^2, \\ &\text{s.t.} \quad \mathbf{A}^T\mathbf{A} = \mathbf{I},\end{aligned} \tag{14}$$

where $\mathbf{P}$ is the projection matrix, w_{ij} is an entry of $\mathbf{W} \in \mathfrak{R}^{n\times n}$ which can be computed by function (7), and α is a regulation parameter to control the trade-off between the global and local structural information. The first term of (14) captures the local structural information of the original data for $\mathbf{P}$ but lacks the global structural information, and the second term of (14) introduces the global Euclidean structural information to $\mathbf{P}$. Finally, the projection matrix $\mathbf{P}$ can obtain the global and local structural information of the original data. Figure 1 illustrates the main idea of the proposed method.

However, the projection matrix $\mathbf{P}$ obtained by (14) is dense and has poor interpretability. To overcome this problem, we employ the $L_{2,1}$-norm penalty. The main reason is that $L_{2,1}$-norm has the property of structured sparsity which helps select the important features of the high-dimensional

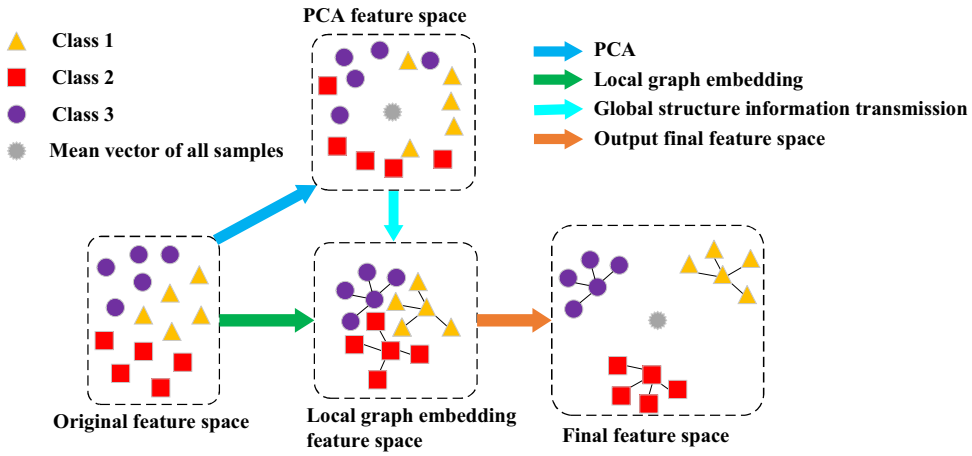

Fig. 1 The illustration of the idea of GSGNPE

data. Hence, the $L_{2,1}$-norm penalty can effectively promote the retention of essential structural information and improve the robustness of our model. Finally, the objective function of GSGNPE is defined as follows:

$$\begin{aligned}\min_{\mathbf{P},\mathbf{A}} &\sum_{i}^{n}\left\|\mathbf{P}^T\mathbf{x}_i-\sum_j w_{ij}\mathbf{P}^T\mathbf{x}_j\right\|_F^2 \\ &+\alpha\sum_{i=1}^{c}||\theta_i-\mathbf{A}^T\mathbf{p}_i||_2^2+\lambda||\mathbf{P}||_{2,1},\quad \text{s.t.}\quad \mathbf{A}^T\mathbf{A}=\mathbf{I},\end{aligned} \tag{15}$$

where λ is the parameter used to adjust the penalty. Compared with other dimensionality reduction methods [33–36] that combine the global and local structure graphs, we adopt a simpler and more effective way to combine global and local structural information instead of constructing complex structure graphs. Our method can also avoid the "out-of-sample" problem. A new sample $\tilde{\mathbf{x}}$ can be explicitly mapped to the low-dimensional space through the projection matrix $\mathbf{P}$ to obtain the low-dimensional representation $\mathbf{P}^T\tilde{\mathbf{x}}$, then using the nearest neighbor classifier to determine the nearest neighbor, and finally we can get the label result based on the nearest neighbor's label. More importantly, the objective function of GSGNPE is concise and easy to optimize.

3.3 Optimization

There are two variables $\mathbf{P}$ and $\mathbf{A}$ in the objective function of GSGNPE, which are difficult to solve directly. We adopt an iterative algorithm to obtain $\mathbf{P}$ and $\mathbf{A}$.

- Update $\mathbf{P}$ by keeping $\mathbf{A}$ fixed

When $\mathbf{A}$ is fixed, the loss function (15) can be written as:

$$\begin{aligned}&\min_{\mathbf{P}}\sum_{i}^{n}\left\|\mathbf{P}^T\mathbf{x}_i-\sum_{j}^{k}\mathbf{W}_{ij}\mathbf{P}^T\mathbf{x}_j\right\|_F^2+\alpha||\mathbf{\Theta}-\mathbf{A}^T\mathbf{P}||_F^2+\lambda||\mathbf{P}||_{2,1}\\ &=\min_{\mathbf{P}} tr(\mathbf{P}^T\mathbf{X}(\mathbf{I}-\mathbf{W})^T(\mathbf{I}-\mathbf{W})\mathbf{X}^T\mathbf{P})\\ &\quad+\alpha||\mathbf{\Theta}-\mathbf{A}^T\mathbf{P}||_F^2+\lambda||\mathbf{P}||_{2,1}\\ &=\min_{\mathbf{P}} tr(\mathbf{P}^T\mathbf{H}\mathbf{P})+\alpha||\mathbf{\Theta}-\mathbf{A}^T\mathbf{P}||_F^2+\lambda||\mathbf{P}||_{2,1},\end{aligned} \tag{16}$$

where $\mathbf{H}=\mathbf{X}(\mathbf{I}-\mathbf{W})^T(\mathbf{I}-\mathbf{W})\mathbf{X}^T$. Take the derivative of (16) with respect to $\mathbf{P}$ and set it to zero, we have:

$$(\mathbf{H}+\mathbf{H}^T)\mathbf{P}+\alpha(-2\mathbf{A}\mathbf{\Theta}+2\mathbf{A}\mathbf{A}^T\mathbf{P})+\lambda\mathbf{Q}\mathbf{P}=0, \tag{17}$$

where $\mathbf{Q}$ is a diagonal matrix with $\mathbf{Q}_{ii}=\frac{1}{||\mathbf{P}^i||_2}$ and $\mathbf{P}^i$ is the ith row vector of $\mathbf{P}$. Denote $(\mathbf{H}+\mathbf{H}^T)$ by $\mathbf{M}$ and we have:

$$\mathbf{P}=2\alpha(\mathbf{M}+2\alpha\mathbf{A}\mathbf{A}^T+\lambda\mathbf{Q})^{-1}\mathbf{A}\mathbf{\Theta} \tag{18}$$

- Update $\mathbf{A}$ by keeping $\mathbf{P}$ fixed

The objective function (15) can be written as:

$$\min_{\mathbf{A}}\sum_{i=1}^{c}||\theta_i-\mathbf{A}^T\mathbf{p}_i||_2^2=\min_{\mathbf{A}}||\mathbf{\Theta}^T-\mathbf{P}^T\mathbf{A}||_F^2,\quad \text{s.t.}\quad \mathbf{A}^T\mathbf{A}=\mathbf{I}. \tag{19}$$

If $\mathbf{A}$ is an orthogonal square matrix with $\mathbf{A}^T\mathbf{A}=\mathbf{A}\mathbf{A}^T=\mathbf{I}$, the above equation can be converted to:

Table 1 The optimization procedure of GSGNPE

Input: Training set $\mathbf{X} = [x_1, x_2, \ldots, x_n] \in \Re^{d\times n}$,the number of neighbors k, the parameters α and λ, the number of iterations *Iter*.

Step 1: Obtain $\mathbf{\Theta}$ by (2);
Step 2: Initialize $\mathbf{P}$ with $\mathbf{\Theta}$;
Initialize $\mathbf{W}$ with the weight matrix by (7);
Step 3: Initialize $\mathbf{A}$ via (22);
Step 4: For $i = 1 : Iter$
Update $\mathbf{P}$ via (18);
Update $\mathbf{A}$ via (22);
End
Output: Projection matrix $\mathbf{P}$, orthogonal matrix $\mathbf{A}$.

$$\begin{aligned}
&\min_{\mathbf{A}} ||\mathbf{\Theta}^T - \mathbf{P}^T\mathbf{A}||_F^2 \\
&= \min_{\mathbf{A}} tr(\mathbf{\Theta}^T\mathbf{\Theta} - 2\mathbf{A}^T\mathbf{P}\mathbf{\Theta}^T + \mathbf{P}^T\mathbf{A}\mathbf{A}^T\mathbf{P}^T) \\
&= \min_{\mathbf{A}} ||\mathbf{\Theta}^T||_F^2 + ||\mathbf{P}^T\mathbf{A}||_F^2 - 2tr(\mathbf{A}^T\mathbf{P}\mathbf{\Theta}^T) \\
&= \min_{\mathbf{A}} ||\mathbf{\Theta}^T||_F^2 + ||\mathbf{P}^T||_F^2 - 2tr(\mathbf{A}^T\mathbf{P}\mathbf{\Theta}^T).
\end{aligned} \tag{20}$$

Thus, (19) is equivalent to the following problem:

$$\max_{\mathbf{A}^T\mathbf{A}=\mathbf{I}} tr(\mathbf{A}^T\mathbf{P}\mathbf{\Theta}^T). \tag{21}$$

Therefore, (19) can be converted into an orthogonal Procrustes problem, by using the conclusion in [40], we perform the singular value decomposition of $\mathbf{P}\mathbf{\Theta}^T$, that is, $\mathbf{P}\mathbf{\Theta}^T = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T$. Then we have:

$$\mathbf{A} = \mathbf{U}\mathbf{V}^T. \tag{22}$$

The detailed optimization procedure of GSGNPE is shown in Table 1.

3.4 Convergence analysis

In this section we will give the convergence analysis of the iterative algorithm in Table 1. Formally, the $L_{2,1}$ norm of the projection matrix $\mathbf{P}$ can be rewritten as:

$$||\mathbf{P}||_{2,1} = 2tr(\mathbf{P}^T\mathbf{S}\mathbf{P}), \tag{23}$$

where $\mathbf{S}$ is a diagonal matrix and the ith diagonal element is denoted as:

$$\mathbf{S}_{ii} = \frac{1}{2||\mathbf{p}^i||_2}, \tag{24}$$

where $\mathbf{p}^i$ is the ith row vector of matrix $\mathbf{P}$. The objective function of GSGNPE can be rewritten as

$$\min_{\mathbf{P},\mathbf{A}} tr(\mathbf{P}^T\mathbf{H}\mathbf{P}) + \alpha||\mathbf{\Theta} - \mathbf{A}^T\mathbf{P}||_F^2 + \lambda' tr(\mathbf{P}^T\mathbf{S}\mathbf{P}), \quad \text{s.t.} \quad \mathbf{A}^T\mathbf{A} = \mathbf{I}, \tag{25}$$

where $\lambda' = 2\lambda$. In order to prove the convergence of the iterative algorithm, we use the following Lemma.

Lemma 1 [41] *For any nonzero vector a and b, the following inequality holds*:

$$||a||_2 - \frac{||a||_2^2}{2||b||_2} \le ||b||_2 - \frac{||b||_2^2}{2||b||_2}. \tag{26}$$

Let $\mathbf{P}_{k+1}$ *and* $\mathbf{P}_k$ *denote the projection matrices in the* $(k+1)$*th and kth iteration, respectively. By using Lemma 1, we can get the following inequality*:

$$||\mathbf{p}_{k+1}^i||_2 - \frac{||\mathbf{p}_{k+1}^i||_2^2}{2||\mathbf{p}_k^i||_2} \le ||\mathbf{p}_k^i||_2 - \frac{||\mathbf{p}_k^i||_2^2}{2||\mathbf{p}_k^i||_2}, \tag{27}$$

where $\mathbf{p}_{k+1}^i$ *and* $\mathbf{p}_k^i$ *are the ith row vectors of* $\mathbf{P}_{k+1}$ *and* $\mathbf{P}_k$*, respectively.*

Proposition 1 *The iterative algorithm in Table 1 decreases the value of the objective function of GSGNPE in each iteration.*

Proof For simplicity, we denote the value of the objective function in the kth iteration as $J(\mathbf{P}_k, \mathbf{S}_k, \mathbf{A}_k)$. Now suppose that $\mathbf{P}_k$, $\mathbf{S}_k$, and $\mathbf{A}_k$ are given. Since $\mathbf{P}_{k+1}$ is updated by (18), the following inequality holds:

$$J(\mathbf{P}_{k+1}, \mathbf{S}_k, \mathbf{A}_k) \le J(\mathbf{P}_k, \mathbf{S}_k, \mathbf{A}_k), \tag{28}$$

According to (25), we can rewrite (28) as:

$$\begin{aligned}
&tr(\mathbf{P}_{k+1}^T\mathbf{H}\mathbf{P}_{k+1}) + \alpha||\mathbf{\Theta} - \mathbf{A}_k^T\mathbf{P}_{k+1}||_F^2 + \lambda' tr(\mathbf{P}_{k+1}^T\mathbf{S}_k\mathbf{P}_{k+1}) \\
&\quad \le tr(\mathbf{P}_k^T\mathbf{H}\mathbf{P}_k) + \alpha||\mathbf{\Theta} - \mathbf{A}_k^T\mathbf{P}_k||_F^2 + \lambda' tr(\mathbf{P}_k^T\mathbf{S}_k\mathbf{P}_k).
\end{aligned} \tag{29}$$

Based on the definition of $L_{2,1}$-norm, we can get the following inequality:

$$
\begin{aligned}
&tr(\mathbf{P}_{k+1}^T\mathbf{H}\mathbf{P}_{k+1}) + \alpha||\mathbf{\Theta} - \mathbf{A}_k^T\mathbf{P}_{k+1}||_F^2 + \lambda^{'}\sum_{i=1}^{d}\frac{||\mathbf{p}_{k+1}^i||_2^2}{2||\mathbf{p}_k^i||_2} \\
&\quad \leq tr(\mathbf{P}_k^T\mathbf{H}\mathbf{P}_k) + \alpha||\mathbf{\Theta} - \mathbf{A}_k^T\mathbf{P}_k||_F^2 + \lambda^{'}\sum_{i=1}^{d}\frac{||\mathbf{p}_k^i||_2^2}{2||\mathbf{p}_k^i||_2}
\end{aligned} \tag{30}
$$

The above inequality can be further transformed into the following inequality:

$$
\begin{aligned}
&tr(\mathbf{P}_{k+1}^T\mathbf{H}\mathbf{P}_{k+1}) + \alpha||\mathbf{\Theta} - \mathbf{A}_k^T\mathbf{P}_{k+1}||_F^2 \\
&\quad + \lambda^{'}\sum_{i=1}^{d}||\mathbf{p}_{k+1}^i||_2 - \left(\lambda^{'}\sum_{i=1}^{d}||\mathbf{p}_{k+1}^i||_2 - \lambda^{'}\sum_{i=1}^{d}\frac{||\mathbf{p}_{k+1}^i||_2^2}{2||\mathbf{p}_k^i||_2}\right) \\
&\leq tr(\mathbf{P}_k^T\mathbf{H}\mathbf{P}_k) + \alpha||\mathbf{\Theta} - \mathbf{A}_k^T\mathbf{P}_k||_F^2 \\
&\quad + \lambda^{'}\sum_{i=1}^{d}||\mathbf{p}_k^i||_2 - \left(\lambda^{'}\sum_{i=1}^{d}||\mathbf{p}_k^i||_2 - \lambda^{'}\sum_{i=1}^{d}\frac{||\mathbf{p}_k^i||_2^2}{2||\mathbf{p}_k^i||_2}\right).
\end{aligned} \tag{31}
$$

It is equivalent to the following inequality:

$$
\begin{aligned}
&tr(\mathbf{P}_{k+1}^T\mathbf{H}\mathbf{P}_{k+1}) + \alpha||\mathbf{\Theta} - \mathbf{A}_k^T\mathbf{P}_{k+1}||_F^2 + \lambda^{'}\sum_{i=1}^{d}||\mathbf{p}_{k+1}^i||_2 \\
&\leq tr(\mathbf{P}_k^T\mathbf{H}\mathbf{P}_k) + \alpha||\mathbf{\Theta} - \mathbf{A}_k^T\mathbf{P}_k||_F^2 + \lambda^{'}\sum_{i=1}^{d}||\mathbf{p}_k^i||_2 \\
&\quad - \left(\left(\lambda^{'}\sum_{i=1}^{d}||\mathbf{p}_k^i||_2 - \lambda^{'}\sum_{i=1}^{d}\frac{||\mathbf{p}_k^i||_2^2}{2||\mathbf{p}_k^i||_2}\right)\right. \\
&\quad \left. -\left(\lambda^{'}\sum_{i=1}^{d}||\mathbf{p}_{k+1}^i||_2 - \lambda^{'}\sum_{i=1}^{d}\frac{||\mathbf{p}_{k+1}^i||_2^2}{2||\mathbf{p}_k^i||_2}\right)\right).
\end{aligned} \tag{32}
$$

By using (27), the following inequality holds:

$$
\begin{aligned}
&tr(\mathbf{P}_{k+1}^T\mathbf{H}\mathbf{P}_{k+1}) + \alpha||\mathbf{\Theta} - \mathbf{A}_k^T\mathbf{P}_{k+1}||_F^2 + \lambda^{'}\sum_{i=1}^{d}||\mathbf{p}_{k+1}^i||_2 \\
&\quad \leq tr(\mathbf{P}_k^T\mathbf{H}\mathbf{P}_k) + \alpha||\mathbf{\Theta} - \mathbf{A}_k^T\mathbf{P}_k||_F^2 + \lambda^{'}\sum_{i=1}^{d}||\mathbf{p}_k^i||_2.
\end{aligned} \tag{33}
$$

According to the definition of $L_{2,1}$-norm and (23), we have the following inequality:

$$
\begin{aligned}
&tr(\mathbf{P}_{k+1}^T\mathbf{H}\mathbf{P}_{k+1}) + \alpha||\mathbf{\Theta} - \mathbf{A}_k^T\mathbf{P}_{k+1}||_F^2 + \lambda^{'}tr(\mathbf{P}_{k+1}^T\mathbf{S}_{k+1}\mathbf{P}_{k+1}) \\
&\quad \leq tr(\mathbf{P}_k^T\mathbf{H}\mathbf{P}_k) + \alpha||\mathbf{\Theta} - \mathbf{A}_k^T\mathbf{P}_k||_F^2 + \lambda^{'}tr(\mathbf{P}_k^T\mathbf{S}_k\mathbf{P}_k).
\end{aligned} \tag{34}
$$

The above inequality is equivalent to

$$
J(\mathbf{P}_{k+1}, \mathbf{S}_{k+1}, \mathbf{A}_k) \leq J(\mathbf{P}_k, \mathbf{S}_k, \mathbf{A}_k). \tag{35}
$$

Since the optimization of $\mathbf{A}_k$ is an orthogonal Procrustes problem, the following inequality holds:

$$
J(\mathbf{P}_{k+1}, \mathbf{S}_{k+1}, \mathbf{A}_{k+1}) \leq J(\mathbf{P}_k, \mathbf{S}_k, \mathbf{A}_k). \tag{36}
$$

□

Proposition 1 shows that the objective function of GSGNPE monotonically decreases in each iteration, and the convergence of our proposed algorithm is thus guaranteed.

4 Experiments

In this section, to demonstrate the effectiveness of our proposed method, GSGNPE is compared with the state-of-the-art methods including PCA, LPP, NPE, UDFS [42], RDR [43], LRPPGRR [44], SOGFS [45], and GLSRGE [35]. The comparison experiments are conducted on ten well-known data sets that are FERET [46], AR [47], CMU PIE [48], extended YaleB [49], The PolyU FKP,[1] binary,[2] and four real benchmark data sets of UCI.[3] For all experiments, we use MATLAB R2020a to run the codes on the machine with an Intel(R) Core(TM)i5-6500 CPU 3.20 GHz and 16 GB RAM.

4.1 Data sets

FERET data set contains 13,539 face images of 1565 individuals, and there is only one individual in each image. Each individual in the images has different illumination, facial expressions, and face orientations. We select a subset from FERET data set, which contains 1400 images of 200 individuals, and each individual has 7 images. All images in the subset are resized to 40×40 pixels.

AR data set consists of 4000 face images of 126 individuals. We selected a subset of 2400 images of 120 individuals from the original data set. The images in selected subset are under different illumination, facial expressions and occlusions. These images are resized to 50×40 pixels.

CMU PIE data set contains 41,368 face images consisting of 68 individuals under different lighting conditions and facial expressions. Our experiments usea selected subset contains 1632 images of 68 individuals, and each individual has 24 images. All images used in our experiments are resized to 32×32 pixels.

Extended YaleB data set used in the experiment contains 2414 frontal cropped facial images of 38 individuals under different illumination and facial expressions. All images of the data set are resized to 32×32 pixels.

The PolyU FKP contains 7920 pictures of fingers of 165 volunteers (125 males and 40 females). We select a subset from PolyU FKP data set, which contains 1980 images of 165 individuals, and each individual has 12 images. All

[1] http://www.comp.polyu.edu.hk/~biometrics/FKP.htm.

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

[3] http://archive.ics.uci.edu/ml.

Table 2 The divisions of all databases

Database	Dimensions	Samples	Classes	l	r
FERET	1600	1400	200	4	3
AR	2000	2400	120	4	16
CMU PIE	1024	1632	68	8	16
Extended YaleB	1024	2414	38	20	34
PolyU FKP	1540	1980	165	4	8
Binary	2116	1404	36	15	14
SCCTS	60	600	6	20	80
IS	19	2310	7	19	311
Isolet	617	1560	2	200	580
Dermatology	34	366	6	15	5–97

PolyU FKP images used in our experiments are resized to 28×55 pixels.

Binary alpha character data set contains 1404 handwriting images of 26 letters (A–Z) and 10 numbers (0–9). Each subject has 39 images with different handwriting. All images in our experiments are resized to 46×46 pixels.

Four real benchmark data sets of UCI including synthetic control chart time series (SCCTS), image segmentation (IS), a subset of ISOLET (Isolet), and dermatology data sets. The detailed information of the data sets is described in Table 2.

Figure 2 shows some images from the data sets used in the experiments. Note that we added the occlusion blocks with different sizes to the four face data sets.

4.2 Experimental settings

All data sets used in the experiments are divided into two parts: training set and test set. For each data set, we randomly selected l images from each class to form the training sets, and the rest r images in each class form the test sets. The detailed settings of all data sets are shown in Table 2. To verify the robustness of GSGNPE, we randomly added some black occlusion blocks with different sizes to each image of the four face data sets including FERET, AR, CMU PIE, and extended YaleB, as shown in Fig. 2a–d. In order to reduce the consumption of computing resources and avoid the problem of small samples caused by high dimensional features, PCA is employed to reduce the dimensions of the original data. More specifically, we selected the first 200 principal components for FERET, AR, CMU PIE, extended YaleB, PolyU FKP, and Isolet. For binary, we only picked the first 50 principal components which could retain most of its information. We do not perform dimensionality reduction on SCCTS, IS, and Dermatology data sets because of their low

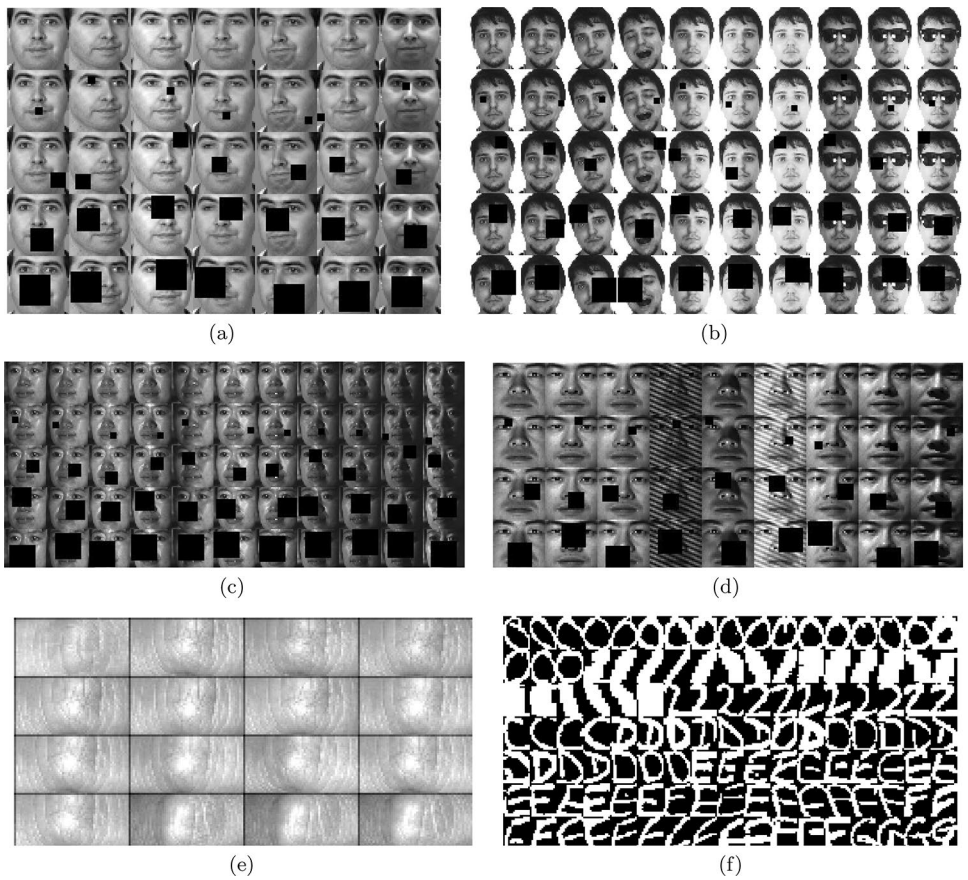

Fig. 2 Some images from the selected data sets. **a** FERET (from top to bottom, occlusion block size: 0×0, 5×5, 10×10, 15×15, 20×20), **b** AR (from top to bottom, occlusion block size: 0×0, 5×5, 10×10, 15×15, 20×20), **c** PIE (from top to bottom, occlusion block size: 0×0, 5×5, 10×10, 15×15, 20×20), **d** extended YaleB (from top to bottom, occlusion block size: 0×0, 5×5, 10×10, 15×15), **e** FKP, **f** binary

Table 3 Best recognition accuracy (%), standard deviation, and dimensions of the comparison methods on FERET

Block size	PCA	LPP	NPE	UDFS	RDR	SOGFS	LRPPGRR	GLSRGE	Our method
0×0	82.75	79.13	79.83	83.22	82.33	83.25	82.72	83.77	**86.78**
	± 15.00	± 18.00	± 18.00	± 15.00	± 16.00	± 15.00	± 15.00	± 14.00	± 12.00
	97	94	99	94	99	97	100	40	99
5×5	79.42	73.92	74.57	78.63	79.52	70.70	79.82	77.25	**80.88**
	± 18.00	± 21.00	± 21.00	± 18.00	± 17.00	± 23.00	± 17.00	± 19.00	± 16.00
	96	87	87	100	97	100	93	26	42
10×10	67.32	64.84	64.93	67.17	66.70	66.02	67.32	67.43	**67.45**
	± 26.00	± 28.00	± 28.00	± 26.00	± 26.00	± 27.00	± 26.00	± 26.00	± 26.00
	94	100	95	82	96	100	98	57	78
15×15	66.40	64.55	64.52	66.28	66.08	65.80	66.37	66.78	**66.98**
	± 27.00	± 28.00	± 28.00	± 27.00	± 27.00	± 27.00	± 27.00	± 26.00	± 26.00
	81	79	96	99	68	100	75	37	47
20×20	65.50	63.95	63.93	65.57	65.33	65.33	65.45	65.97	**66.02**
	± 27.00	± 29.00	± 29.00	± 27.00	± 28.00	± 28.00	± 27.00	± 27.00	± 27.00
	57	73	98	74	99	100	88	28	44

Bold values indicate the highest experimental results in the corresponding experiments

Table 4 Best recognition accuracy (%), standard deviation, and dimensions of the selected methods on AR

Block size	PCA	LPP	NPE	UDFS	RDR	SOGFS	LRPPGRR	GLSRGE	Our method
0×0	81.25	76.9	80.36	78.67	80.34	85.99	81.02	88.10	**89.87**
	± 4.70	± 8.60	± 7.70	± 11.00	± 4.50	± 5.70	± 4.60	± 7.90	± 6.60
	99	100	100	99	100	72	99	69	96
5×5	79.27	73.09	78.16	79.94	78.92	83.15	79.39	83.3	**86.91**
	± 5.30	± 8.20	± 7.10	± 6.90	± 5.40	± 5.00	± 5.20	± 7.40	± 6.50
	99	100	97	92	100	59	100	58	91
10×10	64.97	65.56	69.78	63.03	66.47	72.33	65.04	70.83	**73.26**
	± 5.20	± 7.20	± 7.60	± 4.90	± 5.90	± 5.50	± 5.10	± 7.20	± 6.50
	100	100	99	100	100	54	100	90	95
15×15	47.81	46.32	49.42	53.11	46.88	59.05	47.42	63.47	**65.41**
	± 4.50	± 6.10	± 6.20	± 8.20	± 4.30	± 7.70	± 4.50	± 5.70	± 5.80
	100	100	100	100	100	87	100	93	100
20×20	43.98	39.96	41.95	48.56	43.01	48.32	43.52	57.7	**58.49**
	± 4.60	± 5.30	± 5.50	± 6.30	± 4.50	± 6.90	± 4.60	± 5.50	± 5.10
	100	100	100	99	100	100	100	100	100

Bold values indicate the highest experimental results in the corresponding experiments

dimensionality. The GSGNPE method uses the function (7) to generate the weight matrix W. In order to maintain the consistency of the experiments, the manifold graph matrix in GLSRGE is also constructed by function (7). GSGNPE uses k-nearest neighbor to construct the adjacent graphs, and the effect of the different values of k on the performance of GSGNPE is given in Sect. 4.6. In our experiments, k is set to the number of samples in each class. On each data set, we conducted 10 independent experiments using the nearest neighbor classifier, and their results are averaged.

4.3 Experimental results and analysis

We first conducted the experiments on four face data sets including FERET, AR, CMU PIE, and extended YaleB. Tables 3, 4, 5 and 6 give the average classification accuracy, standard deviation, and corresponding dimensions under the occlusion blocks with different sizes in first 100 dimensions. After that, the experiments were performed on 6 non-face data sets including PolyU FKP, binary, and four UCI data sets, and the results are reported in Table 7. We performed

Table 5 Best classification accuracy (%), standard deviation, and dimensions of the selected methods on CMU PIE

Block size	PCA	LPP	NPE	UDFS	RDR	SOGFS	LRPPGRR	GLSRGE	Our method
0 × 0	86.21	89.29	89.36	89.07	88.53	87.52	85.83	93.38	**93.94**
	± 5.60	± 4.20	± 4.60	± 3.70	± 4.10	± 6.10	± 5.70	± 3.60	± 3.70
	100	98	100	99	100	98	100	56	97
5 × 5	80.29	81.44	84.86	81.03	79.36	83.68	80.32	81.66	**87.19**
	± 5.50	± 5.10	± 5.20	± 2.80	± 4.40	± 6.20	± 5.50	± 2.70	± 2.60
	100	99	100	86	100	100	99	81	99
10 × 10	53.42	55.33	57.31	61.78	52.20	58.37	53.06	64.98	**66.04**
	± 5.70	± 3.90	± 5.10	± 5.80	± 5.60	± 6.2	± 5.70	± 4.50	± 4.10
	100	100	100	100	99	48	100	100	99
15 × 15	45.71	45.25	44.58	49.05	44.27	45.74	45.58	53.26	**54.03**
	± 5.30	± 5.70	± 5.50	± 5.60	± 5.40	± 5.50	± 5.40	± 5.20	± 5.20
	99	100	100	97	100	99	100	100	100
20 × 20	44.08	42.26	41.37	46.73	41.84	44.11	43.86	51.99	**52.38**
	± 5.60	± 5.50	± 5.60	± 6.20	± 5.60	± 5.50	± 5.50	± 5.30	± 4.90
	99	100	100	99	99	100	99	100	99

Bold values indicate the highest experimental results in the corresponding experiments

Table 6 Best classification accuracy (%), standard deviation, and dimensions of the selected methods on extended YaleB

Block size	PCA	LPP	NPE	UDFS	RDR	SOGFS	LRPPGRR	GLSRGE	Our method
0 × 0	64.38	77.88	66.56	77.99	68.54	67.47	63.22	85.91	**86.23**
	± 3.50	± 2.60	± 3.10	± 2.50	± 3.90	± 3.70	± 3.30	± 2.90	± 2.80
	100	100	100	100	100	98	100	98	100
5 × 5	57.48	66.11	60.62	71.30	56.03	61.19	56.75	75.11	**78.71**
	± 3.30	± 2.90	± 2.80	± 3.70	± 2.90	± 4.40	± 3.30	± 2.90	± 3.2
	100	100	100	100	100	99	100	63	100
10 × 10	28.22	35.49	29.53	36.68	26.64	30.64	27.39	50.27	**53.94**
	± 1.50	± 2.60	± 1.40	± 4.30	± 1.90	± 2.50	± 1.50	± 2.50	± 2.20
	100	100	100	100	100	92	100	99	100
15 × 15	18.23	18.98	15.21	29.55	15.46	19.30	17.80	29.41	**31.72**
	± 0.97	± 0.91	± 0.91	± 2.80	± 0.83	± 1.20	± 0.93	± 1.90	± 1.70
	100	100	100	100	100	100	100	93	83

Bold values indicate the highest experimental results in the corresponding experiments

the selected methods on different number of dimensions and repeated the independent experiment 10 times, and the average value of these 10 experiment results was used as the verification result with respect to each dimension. The highest average result of different dimensions in 10 independent repeated experiments was selected as the final evaluation result. The average classification accuracy reported in Tables 3, 4, 5, 6 and 7 is the highest average result over different dimensions. All data sets will be reduced to different dimensions by selected methods. Specifically, the dimensions of four face data sets, FKP and ISO are reduced to 100 dimensions, and the dimension of binary is reduced to 50. Due to the low dimension, SCCTC, IS and Dermatology are respectively projected into the new spaces whose dimensions are the same as the original dimensions. In order to further explore the overall performance of GSGNPE among different dimensions, Tables 8, 9, 10, 11 and 12 report the average performance of all selected methods among different reduced dimensions. The best results among the selected methods are boldfaced in each table. Note that all standard deviations in the tables are calculated for ten classification accuracy under the dimensions corresponding to best average classification accuracy. The standard deviation in Table 3 is relatively large, the possible cause comes from the fact that each class of FERET has a small number of samples but with different characteristics. Each randomly divided training set has different characteristics of samples, hence the standard deviation for the classification accuracy of 10 repeated experiments is relatively large. Figures 3, 4, 5, 6 and 7 show the performance of different methods in different dimensions. Based on the above experimental results, we can draw the following conclusions.

First, in the case of no occlusion, our method achieves the best results on face and non-face data sets, and could

Table 7 Best classification accuracy (%), standard deviation, and dimensions of the selected methods on FKP, binary, and four UCI data sets

Non-face data sets	PCA	LPP	NPE	UDFS	RDR	SOGFS	LRPPGRR	GLSRGE	Our method
FKP	90.60	83.02	86.73	85.83	88.61	89.95	90.30	93.42	**94.42**
	± 3.00	± 4.90	± 4.20	± 5.10	± 3.30	± 3.00	± 3.10	± 2.30	± 2.10
	97	51	76	96	87	100	99	26	32
Binary	78.43	77.65	77.99	78.18	78.22	78.00	78.43	76.74	**78.96**
	± 1.80	± 1.60	± 1.70	± 2.00	± 1.60	± 1.50	± 1.80	± 1.60	± 1.40
	28	36	35	47	33	50	28	17	43
SCCTS	96.83	82.92	94.85	95.40	96.79	95.25	96.83	93.60	**97.12**
	± 1.70	± 2.20	± 2.50	± 1.80	± 1.80	± 2.10	± 1.70	± 1.50	± 1.70
	8	10	7	57	12	60	8	8	9
IS	85.31	83.27	84.11	85.31	85.82	86.50	85.31	83.31	**87.46**
	± 1.70	± 2.40	± 2.50	± 1.70	± 1.90	± 1.70	± 1.70	± 2.80	± 2.20
	13	13	13	19	12	17	13	13	13
Isolet	88.25	81.59	83.01	82.42	86.97	84.17	87.88	86.89	**88.43**
	± 1.20	± 1.70	± 1.50	± 5.40	± 1.20	± 2.10	± 1.40	± 1.20	± 1.10
	100	94	96	98	98	100	95	29	76
Dermatology	93.33	93.66	92.93	93.26	94.49	93.80	93.33	91.09	**95.14**
	± 0.71	± 1.50	± 2.00	± 1.30	± 1.00	± 1.60	± 0.71	± 1.30	± 1.40
	14	19	8	34	19	31	14	7	23

Bold values indicate the highest experimental results in the corresponding experiments

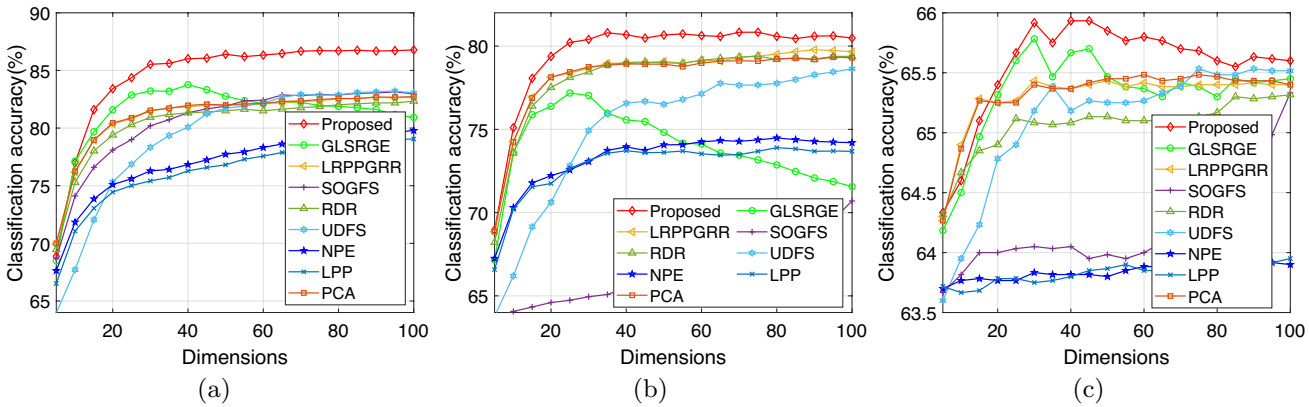

Fig. 3 Average classification accuracy on FERET **a** no occlusion, **b** 5 × 5 occlusion block, and **c** 20 × 20 occlusion block

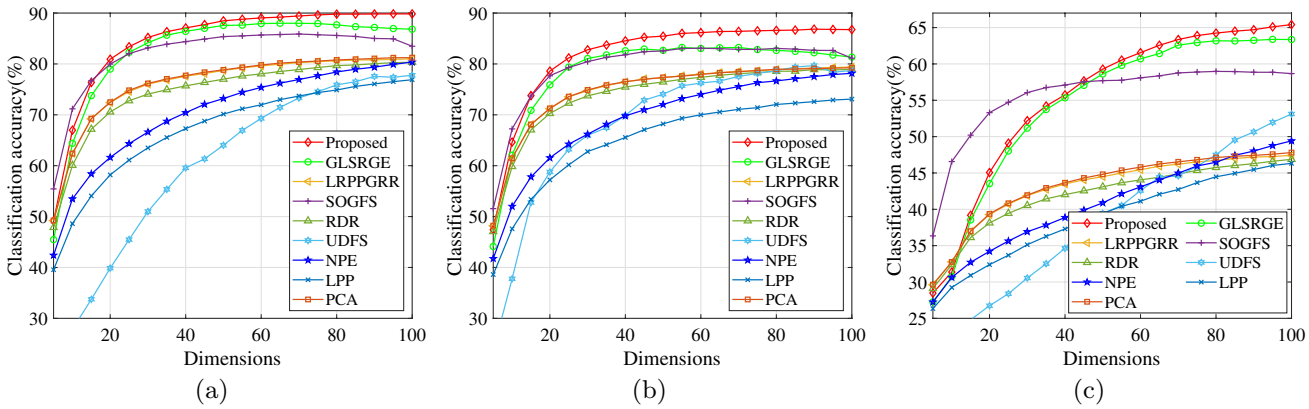

Fig. 4 Average classification accuracy on AR **a** no occlusion, **b** 5 × 5 occlusion block, and **c** 15 × 15 occlusion block

Table 8 Average classification accuracy (%) and standard deviation among different dimensions of the selected methods on FERET

Block size	PCA	LPP	NPE	UDFS	RDR	SOGFS	LRPPGRR	GLSRGE	Our method
0×0	80.64	75.91	76.58	78.84	80.04	80.16	80.61	80.89	**84.12**
	± 3.83	± 3.48	± 3.46	± 5.8	± 3.83	± 4.34	± 3.82	± 4.06	± 5.16
5×5	77.83	72.65	73.05	74.75	77.74	66.34	77.96	73.76	**79.23**
	± 3.16	± 2.14	± 2.25	± 4.56	± 3.3	± 1.86	± 3.22	± 2.62	± 3.57
10×10	66.83	64.53	64.55	66.17	66.15	64.72	66.84	66.61	**66.89**
	± 0.61	± 0.27	± 0.26	± 1.05	± 0.56	± 0.61	± 0.62	± 0.82	± 0.69
15×15	66.04	64.36	64.28	65.66	65.7	64.7	66.02	66.17	**66.43**
	± 0.56	± 0.17	± 0.17	± 0.69	± 0.5	± 0.49	± 0.55	± 0.68	± 0.72
20×20	65.28	63.83	63.83	65.07	65.04	64.18	65.26	65.31	**65.52**
	± 0.38	± 0.09	± 0.07	± 0.58	± 0.31	± 0.38	± 0.37	± 0.44	± 0.51

Bold values indicate the highest experimental results in the corresponding experiments

Table 9 Average classification accuracy (%) and standard deviation among different dimensions of the selected methods on AR

Block size	PCA	LPP	NPE	UDFS	RDR	SOGFS	LRPPGRR	GLSRGE	Our method
0×0	74.87	66.00	69.36	58.96	73.27	80.83	74.73	81.30	**83.01**
	± 10.78	± 11.51	± 11.61	± 18.01	± 10.75	± 10.73	± 10.73	± 13.3	± 12.99
5×5	73.36	64.02	68.23	67.19	72.56	78.14	73.36	77.28	**80.34**
	± 10.46	± 10.74	± 11.26	± 15.99	± 10.63	± 10.74	± 10.47	± 12.3	± 12.58
10×10	57.66	56.64	59.36	50.33	58.78	**67.05**	57.61	60.35	62.36
	± 8.48	± 9.28	± 10.27	± 12.34	± 8.44	± 9.14	± 8.45	± 12.21	± 12.96
15×15	42.79	38.34	40.22	37.86	41.5	**54.90**	42.58	53.61	54.59
	± 5.61	± 6.19	± 6.63	± 9.91	± 5.26	± 7.19	± 5.5	± 11.73	± 11.83
20×20	39.83	34.02	35.51	35.75	38.73	44.74	39.61	48.61	**48.94**
	± 4.65	± 4.31	± 4.71	± 8.36	± 4.3	± 5.26	± 4.52	± 9.48	± 9.45

Bold values indicate the highest experimental results in the corresponding experiments

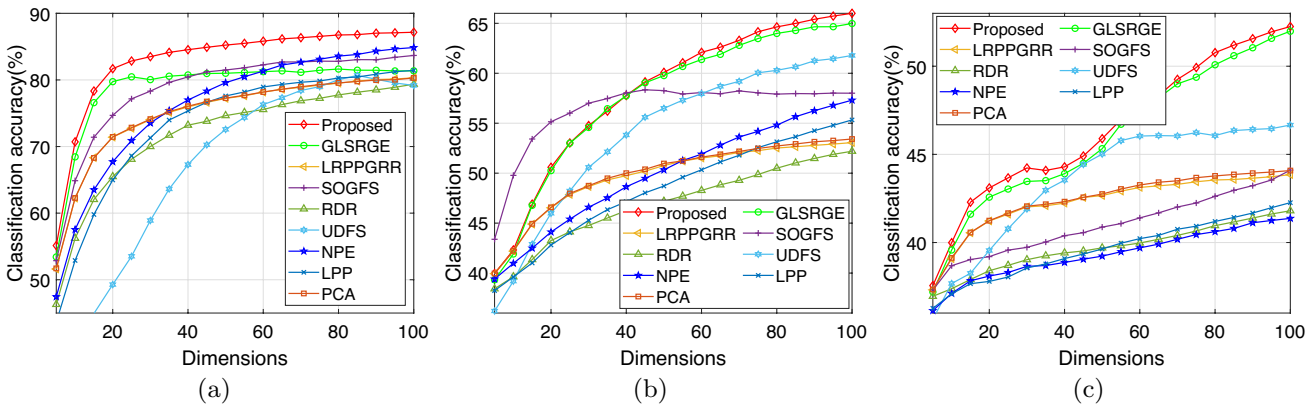

Fig. 5 Average classification accuracy on CMU PIE **a** 5×5 occlusion block, **b** 10×10 occlusion block, and **c** 20×20 occlusion block

obtain stable performance in the subspace with higher dimensions. The potential reason is that preserving the global and local structure of the original data in the low-dimensional space is necessary for dimensionality reduction. The way of combining global and local structure in GSGNPE is more efficient than that in the selected comparison methods.

Second, as the size of occlusion blocks increases, the classification accuracy of all methods decreases gradually, GSGNPE still performs better than comparison methods. GSGNPE integrates global information and $L_{2,1}$-norm, which could alleviate the impact of occlusion blocks.

Third, compared with GSGNPE, GLSRGE has the next-best performance on the four face data sets under the

Table 10 Average classification accuracy (%) and standard deviation among different dimensions of the selected methods on CMU PIE

Block size	PCA	LPP	NPE	UDFS	RDR	SOGFS	LRPPGRR	GLSRGE	Our method
0×0	79.29	79.81	80.36	77.54	77.52	83.3	79.11	89.35	**89.53**
	± 9.83	± 13.51	± 11.46	± 14.21	± 12	± 8.61	± 9.77	± 10.92	± 11.05
5×5	74.00	72.32	75.09	66.25	70.87	77.53	73.96	77.79	**81.75**
	± 8.94	± 11.46	± 11.4	± 14.99	± 9.73	± 9.66	± 8.95	± 9.17	± 10.09
10×10	49.44	47.99	49.51	53.44	46.59	55.89	49.28	56.95	**57.41**
	± 4.16	± 5.34	± 5.6	± 7.96	± 4.15	± 4.7	± 4.08	± 8.17	± 8.27
15×15	43.55	41.34	41.06	45.54	40.85	42.62	43.43	48.48	**48.78**
	± 2.45	± 2.8	± 2.44	± 4.18	± 2.24	± 2.18	± 2.38	± 4.8	± 4.86
20×20	42.23	39.5	39.21	43.36	39.57	40.81	42.1	45.69	**46.14**
	± 2.02	± 1.77	± 1.49	± 3.57	± 1.46	± 1.91	± 1.94	± 4.3	± 4.36

Bold valuesindicate the highest experimental results in the corresponding experiments

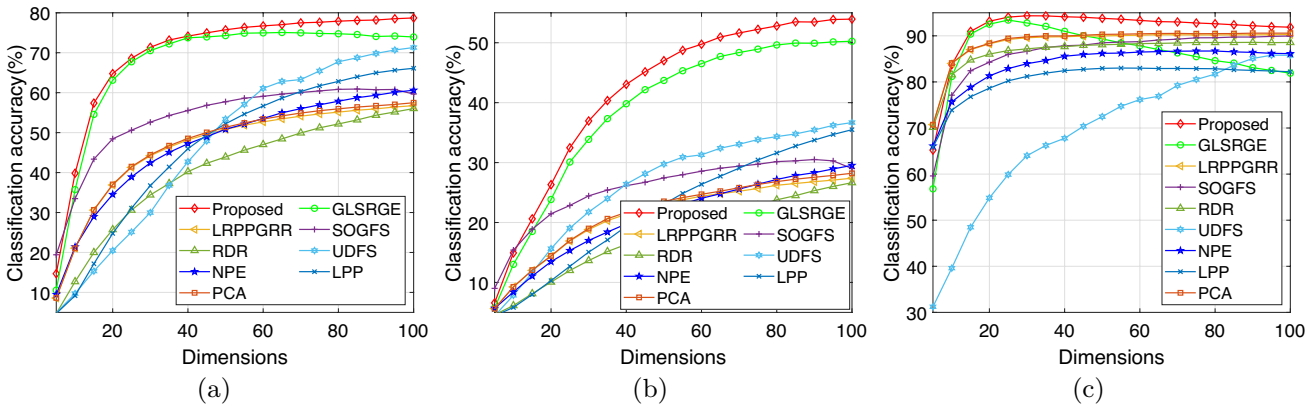

Fig. 6 Average classification accuracy on **a** extended YaleB (5×5 occlusion block), **b** extended YaleB (10×10 occlusion block), and **c** FKP

Table 11 Average classification accuracy (%) and standard deviation among different dimensions of the selected methods on extended YaleB

Block size	PCA	LPP	NPE	UDFS	RDR	SOGFS	LRPPGRR	GLSRGE	Our method
0×0	51.06	54.53	50.95	56.95	46.67	61.06	50.38	75.95	**76.46**
	± 16.17	± 23.85	± 15.93	± 20.53	± 18.91	± 10	± 15.8	± 20.02	± 19.36
5×5	45.59	45.12	45.74	45.88	39.24	52.53	45.12	65.78	**68.00**
	± 14.27	± 19.87	± 14.95	± 22.54	± 15.08	± 12.63	± 14.02	± 18.16	± 18.27
10×10	20.99	21.55	20.36	25.71	17.28	25.17	20.62	37.84	**40.67**
	± 6.91	± 10.06	± 7.28	± 10.1	± 6.81	± 6.34	± 6.67	± 14.11	± 14.96
15×15	14.3	11.98	11.31	20.35	10.6	12.13	14.08	23.01	**24.85**
	± 3.84	± 4.66	± 3.05	± 7.64	± 3.36	± 4.04	± 3.71	± 7.57	± 8.18

Bold values indicate the highest experimental results in the corresponding experiments

occlusion blocks with different sizes, yet it cannot maintain the stable performances in higher dimensions. The possible explanation is that the information of the global structure retained by GSGNPE is more effective than that of GLSRGE, yet the latter constructs a complex global structure graph. In addition, the structured sparsity of L_{21} -norm makes GSGNPE more robust than GLSRGE under the occlusion situations, and also reduces the impact of irrelevant features on retaining global structural information of the original data.

Fourth, the average performance of GSGNPE among different dimensions on all face data sets and most non-face data sets is still better than selected methods. Although the average performance obtained by GSGNPE on AR (with 10×10 and 15×15 occlusion blocks), SCCTS, and IS are not the best, the corresponding results are still competitive, which are almost the second best on these data sets. The

Table 12 Average classification accuracy (%) and standard deviation among different dimensions of the selected methods on FKP, binary and four UCI data sets

Data set	PCA	LPP	NPE	UDFS	RDR	SOGFS	LRPPGRR	GLSRGE	Ourmethod
FKP	87.49	79.58	82.63	68.07	85.4	84.85	87.27	84.81	**89.95**
	± 9.24	± 8.15	± 8.85	± 16.17	± 8.81	± 10.62	± 9.18	± 11.69	± 10.77
Binary	74.23	73.58	73.88	62.62	73.74	72.41	74.24	71.2	**74.82**
	± 8.59	± 8.18	± 8.24	± 16.67	± 9.06	± 8.92	± 8.59	± 8.3	± 8.61
SCCTS	93.76	79.18	92.81	87.98	**94.16**	72.3	93.76	55.84	93.53
	± 6.33	± 4.77	± 5.17	± 12.51	± 5.17	± 16.91	± 6.33	± 16.24	± 6.28
IS	80.32	79.15	72.07	74.15	**82.23**	74.06	80.32	77.47	81.55
	± 9.51	± 6.38	± 18.38	± 13.76	± 8.49	± 11.5	± 9.51	± 11.3	± 10.28
Isolet	85.67	78.85	80.03	72.69	83.44	77.15	85.51	82.13	**85.93**
	± 5.49	± 3.98	± 5.37	± 9.04	± 5.65	± 6.3	± 5.42	± 5.06	± 5.62
Dermatology	90.62	91.35	90.37	80.48	91.52	84.64	90.62	78.77	**92.64**
	± 7.36	± 6.9	± 8.19	± 13.81	± 8.31	± 11.76	± 7.37	± 8.88	± 6.39

Bold values indicate the highest experimental results in the corresponding experiments

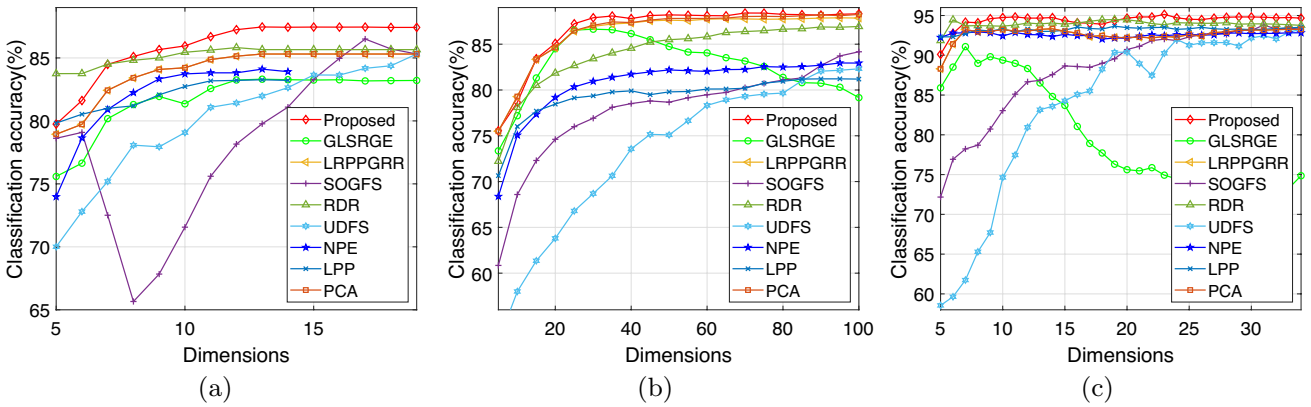

Fig. 7 Average classification accuracy on **a** IS, **b** isolet, and **c** dermatology

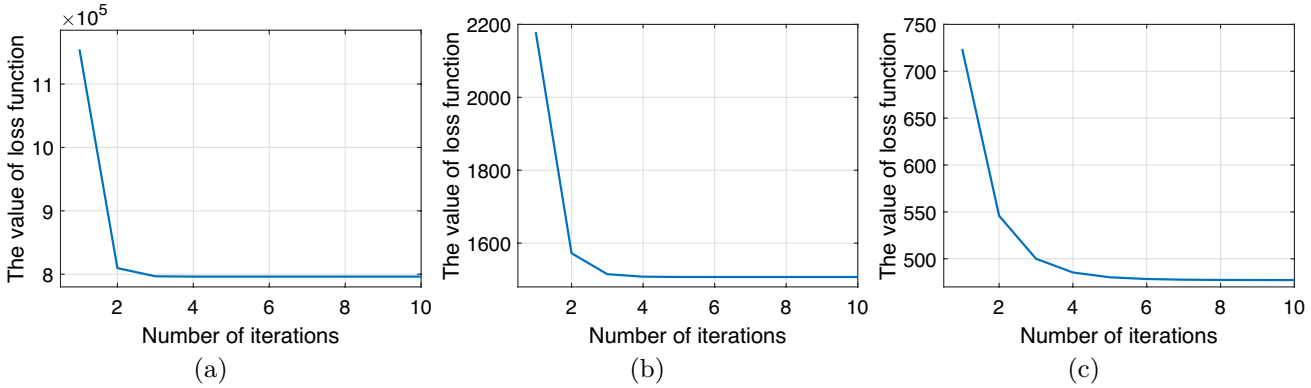

Fig. 8 The trend of the value of the objective function with the different numbers of iterations on **a** PIE (with 20 × 20 occlusion block), **b** SCCTS and **c** IS

overall performance of GSGNPE among different dimensions is better than the selected methods, which further shows the superiority and effectiveness of GSGNPE in keeping the global and local structures in the low-dimensional space.

From the above experimental results, we can conclude that GSGNPE performs better than the comparison methods on both face and non-face data sets. Although GSGNPE retains both global and local structural information in a simple way, it is very efficient. Compared to other

Table 13 The average order value and *Nv* on unoccluded and occluded data sets

Data sets	PCA	LPP	NPE	UDFS	RDR	SOGFS	LRPPGRR	GLSRGE	Our method	*Nv*
Unoccluded data set	4.30	7.00	6.20	5.40	4.60	4.50	4.90	4.40	1.00	3.80
Occluded data set	5.60	6.80	6.73	4.00	7.20	4.60	5.93	2.53	1.00	3.10

selected methods, GSGNPE shows better robustness in the case of occlusion blocks with different sizes and yields better performance under different numbers of dimensions.

To validate the convergence of the proposed iterative algorithm, we recorded the values of the objective function of GSGNPE during the training iterations on three data sets including PIE(with 20×20 occlusion block), SCCTS, and IS. Figure 8 shows the trends of the values of the objective function on three data sets, which indicates that GSGNPE can converge quickly within 10 iterations and verifies our previous conclusion in Proposition 1.

4.4 Statistical significance test

Based on the experimental results obtained, statistical significance tests are conducted for GSGNPE against the comparison methods. We first calculate the average rank of the classification accuracy of each method on the unoccluded and occluded data sets, and then Friedman tests [50] followed by Nemenyi tests [51] are employed. The statistic value F_F in Friedman tests is obtained by the following equation:

$$F_F = \frac{(N-1)\tau_{\chi^2}}{N(K-1)-\tau_{\chi^2}}, \tag{37}$$

where N and K are the numbers of data sets and methods, respectively, and τ_{χ^2} is defined as:

$$\tau_{\chi^2} = \frac{12N}{K(K+1)}\left(\sum_{i=1}^{K} r_i^2 - \frac{K(K+1)^2}{4}\right), \tag{38}$$

where r_i is the average rank of the classification accuracy of the ith method on N data sets. Generally Nemenyi test is adopted to post-hoc tests for Friedman tests. The critical value of Nemenyi test is defined as:

$$Nv = q_\alpha \sqrt{\frac{K(K+1)}{6N}}, \tag{39}$$

where q_α is the critical value of the Turkey distribution. There are 15 occluded data sets with different sizes of occlusion blocks and 10 unoccluded data sets. The number of comparison methods in our experiment is 9. We respectively calculate the F_F values on the unoccluded and occluded data sets, which are 5.54 and 20.37. The critical values of $F(10, 9)$ and $F(15, 9)$ in Friedman test for the significance level $\alpha = 0.05$ are 2.07 and 2.02, which are smaller than 5.54 and 20.37, respectively. This means that the performance differences between these methods are statistically significant. The critical value of the Turkey distribution $q_\alpha = 3.10$ when the significance level $\alpha = 0.05$ and $K = 9$. Then we can obtain the critical values of Nemenyi tests for the unoccluded and occluded data sets, which are 3.80 and 3.10, respectively.

The average rank value of all methods and the critical values of Nemenyi tests are reported in Table 13. Figure 9 is drawn according to Table 13 to show the significant differences between all compared methods. As shown in Fig. 9a, on the unoccluded data sets, there are statistically significant between GSGNPE and the selected methods LRPPGRR, UDFS, NPE, and LPP. However, there is no statistically significant between GSGNPE and the selected methods GLSRGE, SOGFS, RDR, and PCA. When there are occlusion blocks in the data sets, as shown in Fig. 9b, there are statistically significant between GSGNPE and the selected methods LRPPGRR, SOGFS, RDR, NPE, LPP, and PCA, yet, not between GSGNPE and the methods GLSRGE and UDFS. A possible explanation is that the comparison methods in our experiment can retain the structure information when there is no occlusion blocks in data sets, hence they have the similar performance on some cases, resulting in the overlap between their line segments in Friedman test. Nevertheless, compared with other comparison methods, GSGNPE can effectively preserve the structure information of the data sets with occlusion blocks, which makes it have better performance.

4.5 Image visualization

To further explore the ability of GSGNPE to preserve the global and local graph structure information of the original data, we use t-SNE [52] to visualize the distribution of CMU PIE and extended YaleB. SOGFS is a graph embedding based method and also adopts the $L_{2,1}$-norm. Therefore, We employ SOGFS for comparison to prove the effectiveness of our method. Figure 10 indicates the distribution of CMU PIE and extended YaleB before and after projection. We find that compared with SOGFS, GSGNPE can better gather samples with similarities together, which shows that GSGNPE can well retain the local structure of the original data. Furthermore, GSGNPE better keeps different data clusters away from each other in projection space, which proves the effectiveness of GSGNPE in combining global and local structures.

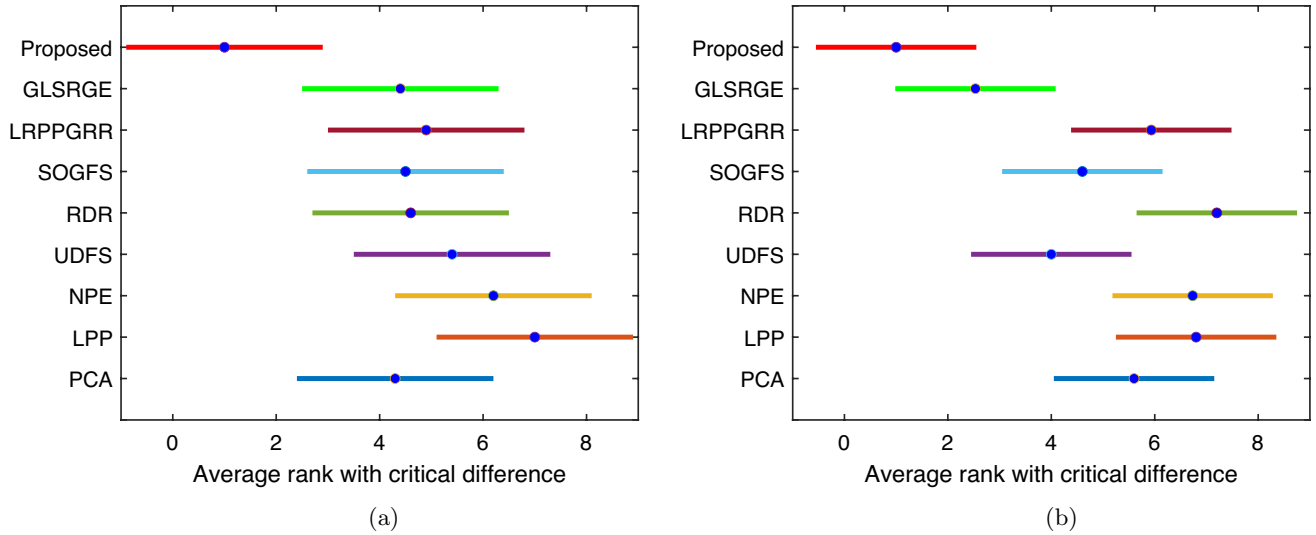

Fig. 9 The significant differences between all compared methods on **a** unoccluded data sets and **b** occluded data sets

4.6 Parameter selection

In this section, we investigate the impacts of the parameters related to GSGNPE including the regulation parameter α and λ, and the number of neighbors k. For all data sets, α and λ are selected from $\{10^{-9}, 10^{-8}, \ldots, 10^{8}, 10^{9}\}$. Figure 11 shows the classification accuracy with different α and λ on AR (no occlusion block), AR (with 5×5 occlusion block), AR (with 10×10 occlusion block), and FKP. As shown in Fig. 11a, in the case of face data set with no occlusion block, when the values of λ and α are in the range of $(10^{7}, 10^{8}]$ and $[10^{-9}, 10^{7}]$ or $[10^{-9}, 10^{7}]$ and $(10^{7}, 10^{8}]$, respectively,

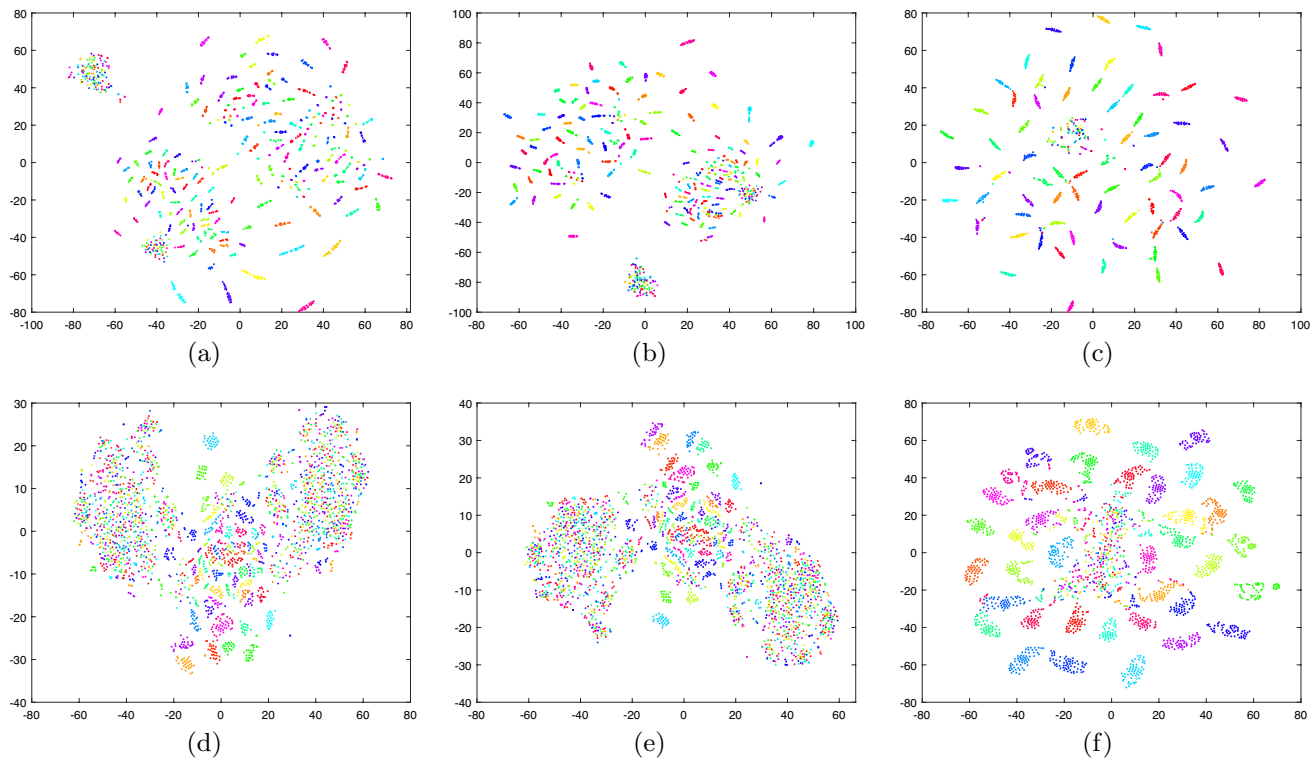

Fig. 10 The t-SNE visualization of distributions of samples. **a** The original samples in CMU PIE, **b** the projected samples by SOGFS in CMU PIE, **c** the projected samples by GSGNPE in CMU PIE, **d** the original samples in extended YaleB, **e** the projected samples by SOGFS in extended YaleB, and **f** the projected samples by GSGNPE in extended YaleB

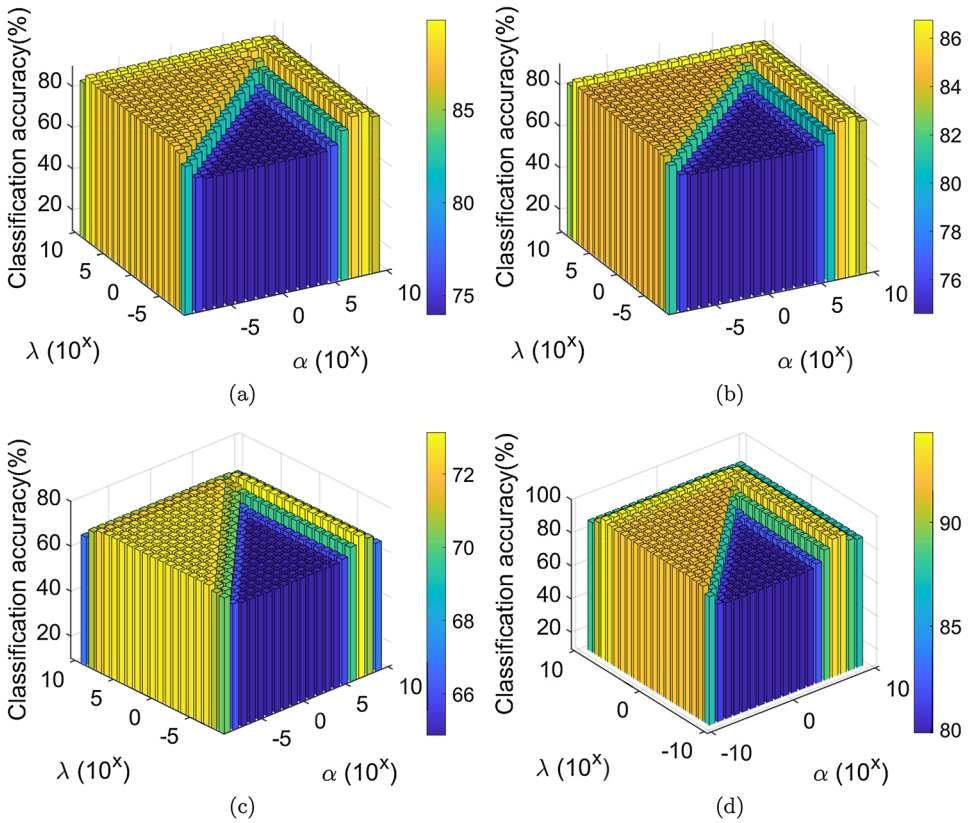

Fig. 11 Classification accuracy with different parameters α and λ on **a** AR (no occlusion block), **b** AR (with 5×5 occlusion block), **c** AR (with 10×10 occlusion block), and **d** FKP

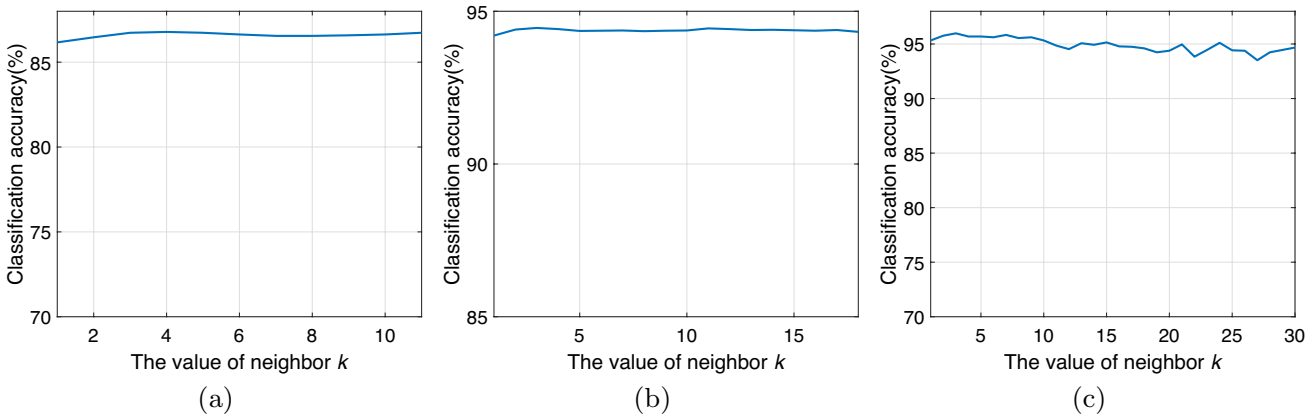

Fig. 12 Classification accuracy with different numbers of neighbors k on **a** FERET, **b** FKP, and **c** dermatology

GSGNPE will perform better. Nevertheless, when there are occlusion blocks on the face data sets, as shown in Fig. 11b and c, α in the range of $[10^7, 10^8]$ and λ smaller than α will usually enable GSGNPE to have the better performance. Moreover, as shown in Fig. 11d, we find that on FKP, the optimal values of λ and α are around 10^7 and $[10^{-9}, 10^7)$ or $(10^6, 10^{-9}]$ and $[10^6, 10^7]$, respectively. Figure 11a–d indicate that the model generally performs better when α is smaller than λ. However, when the value of α is in the range of $[10^7, 10^8]$, λ smaller than α usually makes the model perform well. Based on the above analysis, both λ and α tend to take the large values to make GSGNPE the better performance.

Figure 12 indicates the classification accuracy with different numbers of k on FERET, FKP, and Dermatology. The values of neighbor k in each data set range from 1 to 1.5 times the sample size in each class. We can find that the effectiveness of GSGNPE is not sensitive to k. The possible reason is that the global structural information in GSGNPE is independent of k, which reduces the influence of k in the process of combining local information.

Table 14 Ablation experiments classification accuracy (%), standard deviation, and dimensions

Data set	LSPM	LSPM + GSILM	GSGNPE (LSPM + GSILM + $L_{2,1}$)
FERET (no block)	80.37	85.02	**86.78**
	± 16.00	± 12.00	± 12.00
	38	41	99
FERET (5 × 5 block)	74.25	80.12	**80.88**
	± 21.00	± 16.00	± 16.00
	21	52	42
AR (no block)	84.55	89.04	**89.87**
	± 9.80	± 7.80	± 6.60
	63	88	96
AR (5 × 5 block)	82.02	85.34	**86.91**
	± 8.80	± 7.70	± 6.50
	65	87	91
PIE (no block)	91.26	93.43	**93.94**
	± 3.70	± 3.60	± 3.70
	85	98	97
PIE (5 × 5 block)	83.77	85.61	**87.19**
	± 3.90	± 2.80	± 2.60
	82	100	99
YaleB (no block)	83.53	85.89	**86.23**
	± 3.10	± 2.80	± 2.80
	84	100	100
YaleB (5 × 5 block)	74.28	77.22	**78.71**
	± 3.10	± 3.20	± 3.20
	47	97	100
FKP	89.38	93.47	**94.42**
	± 4.00	± 2.60	± 2.10
	23	32	32
Binary	77.38	78.34	**78.96**
	± 2.00	± 1.20	± 1.40
	17	31	43

Bold values indicate the highest experimental results in the corresponding experiments

4.7 Ablation experiments

We conducted ablation experiments on FERET, AR, PIE, YaleB, FKP, and binary, and the results are shown in Table 14. It can be seen from the experimental results that although the local structure preserving module (LSPM) enables the model to retain the local structure of the original data in the low-dimensional space, only LSPM cannot make the model achieve good performance. However, when the global structure information learning module (GSILM) is added to the model, the performance of the model on the experimental data sets has been improved, which indicates that the model can effectively learn the global structure information of the original data from the existing projection matrix and effectively retain the global and local structure in the low-dimensional space. By adding the $L_{2,1}$-norm to the model, the performance of the model on the experimental data sets has been further improved, which shows that the $L_{2,1}$-norm can effectively promote the model to obtain the essential structural information of the data, learn effectively global structure information, and enhance the robustness of the model. In summary, each part of the final model can effectively enhance the performance of the model and promote the model to retain the essential structural information of the original data in the low-dimensional space.

5 Conclusions

In this paper, we propose a novel graph embedding based method called GSGNPE for unsupervised dimensionality reduction, which is simple and effective. In GSGNPE, the global and local structure of the original high-dimensional data can be effectively retained in the low-dimensional space in a novel way. The projection matrix of PCA rather than the complex global structure graph is employed as the global structural information since PCA could keep the global Euclidean structure in low-dimensional space. A concise yet efficient least-square term is adopted to minimizes the difference between rotated projection matrix and the PCA projection matrix, so as to obtain the global structural information. By combining the objective function of NPE, the global and local structure information of original data can be efficiently combined. The $L_{2,1}$ -term regularization is introduced into GSGNPE to enhance the robustness and improve the interpretability of the obtained projection matrix. Moreover, an iterative optimization algorithm is developed, which could effectively address the optimization problem of GSGNPE and converge fast. Experimental results on face and non-face data sets demonstrate that GSGNPE outperforms the state-of-the-art graph embedding based methods. In the future, it could be worth to construct different local structure graphs and analyze their effect on the performance of GSGNPE. Also, the extended kernel version of GSGNPE for nonlinear complex data is worthy of further research.

Acknowledgements The author would like to thank the Editor-in-Chief, editors and anonymous reviewers for their kind help and valuable comments. The work was supported in part by the National Natural Science Foundation of China (Nos. 61806127, 62076164, 61976145), in part by Guangdong Basic and Applied Basic Research Foundation (No. 2021A1515011861), in part by Shenzhen Institute of Artificial Intelligence and Robotics for Society, in part by Shenzhen Science and Technology Program (No. JCYJ20210324094601005), and in part by the Bureau of Education of Foshan (Nos. 2019XJZZ05).

References

1. Li Y, Chai Y, Yin H et al (2020) A novel feature learning framework for high-dimensional data classification. Int J Mach Learn Cybern. https://doi.org/10.1007/s13042-020-01188-2
2. Hu Q, Zhang L, Zhou Y et al (2018) Large-scale multimodality attribute reduction with multi-kernel fuzzy rough sets. IEEE Trans Fuzzy Syst 26(1):226–238
3. Li J, Mei C, Xu W et al (2015) Concept learning via granular computing: a cognitive viewpoint. Inf Sci 298:447–467
4. Qian J, Yang J, Xu Y et al (2020) Image decomposition based matrix regression with applications to robust face recognition. Pattern Recognit 102:107204
5. Shang R, Chang J, Jiao L et al (2019) Unsupervised feature selection based on self-representation sparse regression and local similarity preserving. Int J Mach Learn Cybern 10:757–770
6. Wang X, Dong L, Yan J (2012) Maximum ambiguity-based sample selection in fuzzy decision tree induction. IEEE Trans Knowl Data Eng 24:1491–1505
7. Wu W, Qian Y, Li T et al (2017) On rule acquisition in incomplete multi-scale decision tables. Inf Sci 378:282–302
8. Shahdoosti H, Tabatabaei Z (2020) Object-based feature extraction for hyperspectral data using firefly algorithm. Int J Mach Learn Cybern 11:1277–1291
9. Fang X, Teng S, Lai Z et al (2018) Robust latent subspace learning for image classification. IEEE Trans Neural Netw Learn Syst 29(6):2502–2515
10. Wang X, He Y (2016) Learning from uncertainty for big data: future analytical challenges and strategies. IEEE Syst Man Cybern Mag 2:26–31
11. Qian J, Yang J, Tai Y et al (2016) Exploring deep gradient information for biometric image feature representation. Neurocomputing 213:162–171
12. Ma M, Deng T, Wang N et al (2019) Semi-supervised rough fuzzy Laplacian eigenmaps for dimensionality reduction. Int J Mach Learn Cybern 10:397–411
13. Wold S, Esbensen K, Geladi P (1987) Principal component analysis. Chemom Intell Lab Syst 2(1–3):37–52
14. Belhumeur P, Hespanha J, Kriegman D (1997) Eigenfaces vs. fisherfaces: recognition using class specific linear projection. IEEE Trans Pattern Anal Mach Intell 19(7):711–720
15. Cai D, He X, Zhou K et al (2007) Locality sensitive discriminant analysis. In: Proceedings of 2007 international joint conference on artificial intelligence (IJCAI07), pp 1713–1726
16. Park S, Kwak N (2018) Independent component analysis by lp-norm optimization. Pattern Recognit 76:752–760
17. Mi J, Zhang Y, Li Y et al (2020) Generalized two-dimensional PCA based on $\ell_{2,p}$-norm minimization. Int J Mach Learn Cybern 11:2421–2438
18. Hu Q, Zhang S, Xie Z et al (2014) Noise model based v-support vector regression with its application to short-term wind speed forecasting. Neural Netw 57:1–11
19. Lai Z, Bao J, Kong H et al (2020) Discriminative low-rank projection for robust subspace learning. Int J Mach Learn Cybern 11:2247–2260
20. Jenssen R (2010) Kernel entropy component analysis. IEEE Trans Pattern Anal Mach Intell 32(5):847–860
21. Xiong F, Gou M, Camps O et al (2014) Person re-identification using kernel-based metric learning methods. In: Proceedings of the European conference on computer vision, pp 1–16
22. Li H, Jiang T, Zhang K (2004) Efficient and robust feature extraction by maximum margin criterion. IEEE Trans Neural Netw 17(1):157–165
23. Tenenbaum J, De Silva V, Langford J (2000) A global geometric framework for nonlinear dimensionality reduction. Science 290(5500):2319–2323
24. Roweis S, Saul L (2000) Nonlinear dimensionality reduction by locally linear embedding. Science 290(5500):2323–2326
25. Belkin M, Niyogi P (2003) Laplacian eigenmaps for dimensionality reduction and data representation. Neural Comput 15(6):1373–1396
26. Fang X, Xu Y, Li X et al (2018) Regularized label relaxation linear regression. IEEE Trans Neural Netw Learn Syst 29(4):1006–1018
27. He X, Niyogi P (2003) Locality preserving projections. In: Proceedings of the 16th international conference on neural information processing systems, pp 153–160
28. He X, Cai D, Yan S et al (2005) Neighborhood preserving embedding. In: Proceedings of the tenth IEEE international conference on computer vision (ICCV05), pp 1208–1213
29. Pang Y, Zhang L, Liu Z et al (2005) Neighborhood preserving projections (NPP): a novel linear dimension reduction method. In: Proceedings of international conference on intelligent computing, pp 117–125
30. Qiao L, Chen S, Tan X (2010) Sparsity preserving projections with applications to face recognition. Pattern Recognit 43(1):331–341
31. Cai W (2017) A dimension reduction algorithm preserving both global and local clustering structure. Knowl Based Syst 118:191–203
32. Fang X, Han N, Wong W et al (2019) Flexible affinity matrix learning for unsupervised and semisupervised classification. IEEE Trans Neural Netw Learn Syst 30(4):1133–1149
33. Yin M, Gao J, Lin Z (2016) Laplacian regularized low-rank representation and its applications. IEEE Trans Pattern Anal Mach Intell 38(3):504–517
34. Liu Z, Shi K, Zhang K et al (2020) Discriminative sparse embedding based on adaptive graph for dimension reduction. Eng Appl Artif Intell 94:103758
35. Shen X, Liu S, Bao B et al (2020) A generalized least-squares approach regularized with graph embedding for dimensionality reduction. Pattern Recognit 98:107023
36. Gou J, Yi Z, Zhang D et al (2018) Sparsity and geometry preserving graph embedding for dimensionality reduction. IEEE Access 6:75748–75766
37. Zou H, Hastie T, Tibshirani R (2006) Sparse principal component analysis. J Comput Graph Stat 15(2):265–286
38. Hu Q, Li L, Zhu P (2013) Exploring neighborhood structures with neighborhood rough sets in classification learning. In: Rough Sets and Intelligent Systems-Professor Zdzisław Pawlak in Memoriam, Springer, pp 277–307
39. Qian J, Yang J, Zhang N et al (2014) Histogram of visual words based on locally adaptive regression kernels descriptors for image feature extraction. Neurocomputing 129:516–527
40. Golub G, Van Loan C (1996) Matrix computations. Johns Hopkins University Press, Baltimore
41. Nie F, Huang H, Cai X et al (2010) Efficient and robust feature selection via joint $\ell_{2,1}$-norms minimization. In: Advances in neural information processing systems, pp 1813–1821
42. Yang Y, Shen H, Ma Z et al (2011) $\ell_{2,1}$-norm regularized discriminative feature selection for unsupervised learning. In: Proceedings of the 22nd international joint conference on artificial intelligence, pp 1589–1594
43. Lai Z, Mo D, Wong W et al (2018) Robust discriminant regression for feature extraction. IEEE Trans Cybern 48(8):2472–2484
44. Wen J, Han N, Fang X et al (2019) Low-rank preserving projection via graph regularized reconstruction. IEEE Trans Cybern 49(4):1279–1291

45. Nie F, Zhu W, Li X (2019) Structured graph optimization for unsupervised feature selection. IEEE Trans Knowl Data Eng 33(3):1210–1222
46. Phillips P, Moon H, Rizvi S et al (2000) The FERET evaluation methodology for face-recognition algorithms. IEEE Trans Pattern Anal Mach Intell 22(10):1090–1104
47. Martinez A (1998) The AR face database. CVC Tech. Report#24
48. Sim T, Baker S, Bsat M (2002) The CMU pose, illumination, and expression (PIE) database. In: Proceedings of fifth IEEE international conference on automatic face gesture recognition, pp 53–58
49. Georghiades A, Belhumeur P, Kriegman D (2001) From few to many: illumination cone models for face recognition under variable lighting and pose. IEEE Trans Pattern Anal Mach Intell 23(6):643–660
50. Pohlert T (2014) The pairwise multiple comparison of mean ranks package (PMCMR). R Packag 27(2020):10
51. Benavoli A, Corani G, Mangili F (2016) Should we really use post-hoc tests based on mean-ranks. J Mach Learn Res 17(1):152–161
52. van der Maaten L, Hinton G (2008) Visualizing data using t-SNE. J Mach Learn Res 9(2605):2579–2605