Multimedia Tools and Applications (2020) 79:29179–29198
https://doi.org/10.1007/s11042-020-09392-6

Face clustering via learning a sparsity preserving low-rank graph

Changpeng Wang[1] · Jiangshe Zhang[2] · Xueli Song[1] · Tianjun Wu[1]

Received: 12 October 2019 / Revised: 22 May 2020 / Accepted: 21 July 2020 /
Published online: 9 August 2020

Abstract

Face clustering aims to group the face images without any label information into clusters, and has recently attracted considerable attention in machine learning and data mining. Many graph based clustering methods have been proposed and among which sparse representation (SR) and low-rank representation (LRR) are two representative methods for affinity graph construction. The clustering result may be inaccurate if the affinity graph is constructed with low quality. In this paper, we propose a novel face clustering method via learning a sparsity preserving low-rank graph (LSPLRG), where the initial affinity graph is derived on the sparse coefficients without any a priori graph or similarity matrix. In addition, an adaptive weighted matrix is imposed on the data reconstruction errors to enhance the role of important features, while a constraint on the representation matrix is to reduce the redundant features. By integrating the local distance regularization term into LRR, LSPLRG could exploit the global and local structures of data simultaneously. These appealing properties allow LSPLRG to well capture the intrinsic structure of data, and thus has potential to improve clustering performance. Experiments conducted on several face image databases demonstrate the effectiveness and robustness of LSPLRG compared with several state-of-the-art subspace clustering methods.

Keywords Low-rank representation · Graph learning · Face clustering

1 Introduction

Clustering is a fundamentally important task to numerous applications, such as image classification [34], saliency detection [40], image segmentation [44] and motion segmentation [31]. The goal of clustering is to simultaneously segment unlabeled data points into clusters so that the data points in the same clusters are more similar to each other than those

✉ Xueli Song
xlsung@sina.com

[1] School of Science, Chang'an University, Xi'an, 710064, China

[2] School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an, 710049, China

in different clusters [38]. Under the Lambertian assumption, the face images of a subject with a fixed pose and varying illumination approximately lie in a linear subspace of dimension 9 [1]. Thus, the face clustering problem can be considered as image clustering problem over a union of subspaces. In the past decades, a large number of clustering methods have emerged, such as k-means [26], spectral clustering [28], support vector clustering [2], maximum margin clustering [37] and multi-view subspace clustering [42, 48, 49].

In the big data era, high-dimensional data is ubiquitous in many real applications such as image processing [18, 20]. However, high-dimensional data not only results in high computational cost of time and memory for related algorithms, but also degrades their performances due to the inevitable noise and insufficient number of training samples [17, 19, 27]. Under the assumption that high-dimensional data almost lie in multiple low-dimensional subspaces, subspace clustering has gained a lot of attention due to its capability and efficiency in data clustering [33]. The key idea in subspace clustering is to construct a weighted affinity graph matrix from the initial database. Thus, constructing an informative graph to capture the essential relationship of data plays a key role for data clustering.

Many research studies on graph learning investigate how to better capture the intrinsic structure information of data [41, 43, 46]. Generally speaking, the affinity graph can be constructed based on pairwise distance (e.g. Euclidean distance) or reconstruction coefficients. Recent methods for learning the affinity graph are based on the self-expressiveness property, which states a sample in a union of subspaces can be expressed as a linear combination of other samples, i.e., $X = XZ$. Furthermore, a symmetric affinity graph matrix is induced from the new representation Z (i.e. $G = \frac{1}{2}(|Z| + |Z^T|)$), where $G_{ij} = G_{ji}$ determines samples i and j belong to the same subspace. The main issue of existing graph learning methods is that all features are treated equally in the graph construction even if many features are redundant features or even noises.

Recently, sparse representation and low-rank representation have attracted much attention in data clustering due to their success in adaptively exploit the intrinsic representation structures of data [15, 30]. We try to combine their advantages and address the issue in the existing adaptive graph learning methods. In this paper, we propose a novel face clustering method via learning a sparsity preserving low-rank graph (LSPLRG). Specifically, the first step is to learn a sparse affinity graph by data self-representativeness and ℓ_1-norm. Then, a weighted matrix on the data reconstruction errors is imposed to reduce the useless features with large reconstruction errors. Moreover, a distance regularization term is imposed to preserve local structure information of the data, and a constraint on the representation matrix is added in the objective function to alleviate the relevance. These meaningful factors could improve the effectiveness of the proposed model and encourage us to learn a graph to reveal the intrinsic similarity relationships of samples.

The main contributions of this paper are listed as follows:

1. The proposed method learns a initial affinity graph on the sparse coefficients without any a priori graph or similarity matrix. The sparsity could make the obtained graph better capture the intrinsic structure of the data when they are suffered from noise.
2. An adaptive weighted matrix is imposed on the data reconstruction errors to enhance the role of important features, while a constraint on the representation matrix is to reduce the redundant features.
3. The proposed model could exploit the global and local structure of data by LRR and distance regularization term, which ensures to learn a more effective graph for face clustering.

The rest of this paper is organized as follows. Section 2 reviews the related works including low-rank representation and sparse subspace clustering. In Section 3, we introduce the details of our proposed clustering method LSPLRG and its optimizing schemes. Experimental results are presented for illustration in Sections 4 and 5 concludes this paper.

2 Related works

In this section, we briefly review the related work, such as low-rank representation (LRR) and sparse subspace clustering (SSC) before introducing our model. Before reviewing the related work, we define some notations. For a matrix X, x_j is its ith column and x_{ij} is its (i, j)th entry. The Frobenius norm and nuclear norm are denoted by $\|X\|_F$ and $\|X\|_*$ (the sum of the singular values of X), respectively. $\odot$ denotes the element-wise multiplication. X^T is the transpose of X and $tr(X)$ is the trace of X.

2.1 Low-rank representation

Recently, theoretical advances on LRR enable us to explore low-dimensional subspace structures embedded in data. Given a set of data, LRR aims at finding the lowest-rank representation of all data jointly and preserving the membership of samples that belong to the same subspace [23]. Thus, the data usually can be represented by other data that lie in the same subspace when the subspace are independent and the data is noiseless. Generally, the LRR problem can be formulated as follows:

$$\min_{Z} \|Z\|_* \quad \text{subject to} \quad X = AZ \tag{1}$$

where the columns of A are a set of known bases or dictionary items and Z is called the low-rank representation of the data X. $\|Z\|_*$ is the nuclear norm, which is the convex envelope of the rank function [7].

In the real world applications, observation data often contain noise corruption, and data matrix X itself is used as the dictionary. With the balance parameter λ, a more general model version of (1) can be presented as follows:

$$\min_{Z} \|Z\|_* + \lambda \|E\|_l \quad \text{s.t.} \quad X = XZ + E \tag{2}$$

Here, there are many strategies to define the error term E. For example, ℓ_0-norm characterizes the random corruption, $\ell_{2,1}$-norm generally characterizes sample-specific corruption, and F-norm is proposed for the small Gaussian noise. A number of methods have been proposed for solving the above low-rank matrix problems, and the most commonly used methods are Augmented Lagrange Multiplier (ALM) and its variants. The advantage of LRR mainly comes from the low-rank component could reduce the influence of the outliers, which means LRR has the ability to correct the corruptions in data automatically.

2.2 Sparse subspace clustering

In recent years, Sparse Representation (SR) [10] has attracted much attention due to its effectiveness for representing and compressing high-dimensional signals. According to Compressed Sensing (CS) theory [4], the minimum ℓ_1-norm solution to an underdetermined system of linear equation is also the sparsest possible solution under general conditions. Inspired by SR, the standard sparse subspace clustering (SSC) algorithm [13] has been proposed to cluster data points that lie in a union of low-dimensional subspace. By exploiting the self-expressiveness property of the data X, the formulation of SSC is

$$\min_{Z} \|Z\|_1 + \lambda \|E\|_F \quad \text{s.t.} \quad X = XZ + E, \quad diag(Z) = 0 \tag{3}$$

where Z is the coefficient matrix. Each column of Z is the sparse representation vector corresponding to each data point. E is the representation error and λ is a tradeoff parameter.

The main difference between LRR and SSC is the choice of the regularization term of Z. As can be seen from problem (3), $\|Z\|_1$ is used as a convex surrogate of $\|Z\|_0$ to promote sparsity of Z in SSC, and $\|Z\|_*$ is used to seek a jointly low-rank representation of all data in LRR. The element Z_{ij} in Z reflects the similarity between data pair x_i and x_j. Hence Z is often used to define the affinity matrix $(|Z| + |Z^T|)/2$ for final segmentation of the data. The clustering results are obtained by applying a spectral clustering algorithm [25], such as normalized cuts (NCuts) [32].

3 Learning a sparsity preserving low-rank graph

In the clustering task, we have a set of unlabeled data $X = [x_1, x_2, ..., x_n] \in \mathbb{R}^{d \times n}$ and aim to group the data into c clusters. An effective affinity graph which could capture the intrinsic structures of data is essential to obtain better clustering performance. In this section, we present the details of our algorithm including graph initialization and graph learning. Then, an optimization scheme based on iterative updating rules is used to solve the objective function.

3.1 Graph initialization

Recently, sparse coding becomes a widely adopted tool which supposes that any signal can be composed by some basic signals. Different from the other graph learning methods that learning a initial graph by the distances of data points, we attempt to learn a initial graph by the sparse representation of data points. To construct a sparse graph, the objective function is to calculate a representation matrix $Z = [z_1, z_2, ..., z_n]$, which is the solution to ℓ_1 problem

$$\arg\min_{Z} \|X - XZ\|_F^2 + \lambda \|Z\|_1 \tag{4}$$

where $\|\cdot\|_F$ is the Frobenius norm of the matrix and $\|Z\|_1$ is the ℓ_1 norm of the matrix Z. The coefficient matrix Z can be regarded as an asymmetric graph matrix for a dictionary learning problem in which the dictionary is already given by the data themselves. For each data x_i, the vector z_i denotes the sparse coding vector required for constructing the sample x_i from the set of data points. As can be seen, the optimization of problem (4) seeks a sparse graph matrix with a smaller reconstruction error. The sparsity could make the obtained graph better capture the intrinsic structure of the data points when they are suffered from noises.

3.2 Graph learning

In the task of clustering, the learned similarity graph should match the true affinity between data points, and capture the multi-subspaces structure information. To this end, many researchers proposed to impose the graph regularization term in order to learn an affinity graph with high quality [6, 11, 35], and these methods are proved to be effective under mixed conditions.

However, there is a severe problem that many methods treat the reconstruction errors equally in the linear representation, which is harmful to capture the intrinsic structure of data. A robust graph learning method should assign different weight adaptively on the reconstruction error to reinforce its effect during graph learning. Specifically, larger reconstruction error should be assigned smaller weight and important feature should be assigned larger weight respectively. Thus, a weighted nonnegative low-rank representation framework can be defined as

$$\begin{aligned} &\min_{W,Z} \|W^{\frac{1}{2}} \odot E\|_F^2 + \frac{\lambda_1}{2}\|W\|_F^2 + \lambda_2\|Z\|_* \\ &\text{s.t.} \quad X = XZ + E,\, Z,\, W \geqslant 0,\, W^T\mathbf{1} = \mathbf{1} \end{aligned} \tag{5}$$

where W is the weighted matrix with positive values of all elements, and $W^{\frac{1}{2}}$ is defined as an element-wise square root of W. The constraint term $W^T\mathbf{1}$ ($\mathbf{1} \in \mathbb{R}^{d\times 1}$ is a vector that all elements are 1) ensures the weight treats all samples equally. The second term in this criterion is a regularization term given by the Frobenius norm, which provides a solution with the majority of elements are not null. Minimizing the optimization sub-problem to variable W, i.e., $\min_{W\geqslant 0, W^T\mathbf{1}=\mathbf{1}} \|W^{\frac{1}{2}} \odot E\|_F^2 + \frac{\lambda_1}{2}\|W\|_F^2$ leads to a sparse weighted matrix [36]. Boundary constraints $W \geqslant 0$, $W^T\mathbf{1} = \mathbf{1}$ can avoid trivial solution [47] and $Z \geqslant 0$ ensures the learned graph is interpretable and its each element reveals the similar degree of the corresponding two samples.

Beyond low rank property, local similarity structure learned from data points is proved to be very helpful for subspace clustering [22]. Intuitively, close (similar) data points should have close (similar) representation coefficients. In order to exploit the local relationship between the data points, a distance regularization term to constrain the affinity matrix Z is imposed in the subspace clustering. Then the model can be written as

$$\begin{aligned} &\min_{W,Z} \|W^{\frac{1}{2}} \odot E\|_F^2 + \frac{\lambda_1}{2}\|W\|_F^2 + \lambda_2\|Z\|_* + \lambda_3 Tr(D^T Z) \\ &\text{s.t.} \quad X = XZ + E,\, Z,\, W \geqslant 0,\, W^T\mathbf{1} = \mathbf{1} \end{aligned} \tag{6}$$

where parameter λ_3 is set to control the weight of corresponding regularizing term and $Tr(\cdot)$ is the trace operation. By the definition that the ith row and jth column element of matrix D is $d_{ij} = \|x_i - x_j\|_2^2$, then $Tr(D^T Z) = \sum_{i,j=1}^n d_{ij} z_{ij}$. Thus, the third term enforces that the samples with small distance should have similar representations. By imposing the distance regularization term, the local relationship between the points is preserved such that the clustering performance can be improved.

Once obtaining the representation matrix Z, the graph learning methods usually obtain the similarity affinity matrix from Z. In order to alleviate the relevance among the rows of Z, a constraint on Z is added in the objective function. To avoid the sample is selected to represent itself and the trivial solution, we constrain the affinity matrix Z such that the values of its diagonal elements are zero and the sum of its each row is one. Our final graph learning model of LSPLRG can be formulated as follows

$$\begin{aligned} &\min_{W,Z} \|W^{\frac{1}{2}} \odot E\|_F^2 + \frac{\lambda_1}{2}\|W\|_F^2 + \lambda_2\|Z\|_* + \lambda_3 Tr(D^T Z) + \lambda_4 Tr(Z^T L_e Z) \\ &\text{s.t.} \quad X = XZ + E,\, Z,\, W \geqslant 0,\, W^T\mathbf{1} = \mathbf{1},\, diag(Z) = 0,\, Z\bar{\mathbf{1}} = \bar{\mathbf{1}} \end{aligned} \tag{7}$$

where $\bar{\mathbf{1}} \in \mathbb{R}^{n\times 1}$ and $L_e = \bar{\mathbf{1}}\bar{\mathbf{1}}^T - I$ (I is identity matrix). In the optimization, we propose an alternative method to update each variable. Thus, both the matrix factorization and the graph learning are achieved in the learned subspace, and the merit of subspace clustering is inherited.

3.3 Update rules

To find the solutions to (7), we use the alternating direction method of multipliers (ADMM) [3] to obtain the local optimal solution of variables W and Z. We first introduce two auxiliary variables $E = X - XZ$ and $U = Z$ to make the optimization problem (7) separable, and problem (7) is rewritten as

$$\begin{aligned} &\min_{W,Z} \|W^{\frac{1}{2}} \odot E\|_F^2 + \frac{\lambda_1}{2}\|W\|_F^2 + \lambda_2\|U\|_* + \lambda_3 Tr(D^T Z) + \lambda_4 Tr(Z^T L_e Z) \\ &\text{s.t.} \quad X = XZ + E, Z = U, Z, W \geqslant 0, W^T\mathbf{1} = \mathbf{1}, diag(Z) = 0 \end{aligned} \tag{8}$$

The corresponding augmented Lagrangian function of (8) is as follows

$$\begin{aligned} L(Z, W, E, U, Y_1, Y_2) = &\|W^{\frac{1}{2}} \odot E\|_F^2 + \frac{\lambda_1}{2}\|W\|_F^2 + \lambda_2\|U\|_* \\ &+\lambda_3 Tr(D^T Z) + \lambda_4 Tr(Z^T L_e Z) \\ &+\frac{\mu}{2}\left(\|X - XZ - E - \frac{Y_1}{\mu}\|_F^2 + \|Z - U - \frac{Y_2}{\mu}\|_F^2\right) \end{aligned} \tag{9}$$

where Y_1 and Y_2 are Lagrangian multipliers, and μ is a scalar parameter. The augmented Lagrangian is separable and can be minimized along one coordinate direction at each iteration, i.e. minimizing the augmented Lagrangian with respect to one variable alternately with others being fixed. We introduce the detailed procedures and the solution of each subproblem in the following

Update W: Fix the other variables and update W by solving the following problem

$$\min_{W, W^T\mathbf{1}=\mathbf{1}} \|W^{\frac{1}{2}} \odot E\|_F^2 + \frac{\lambda_1}{2}\|W\|_F^2 \tag{10}$$

which can be updated by the element-wise strategy. Obviously, problem (10) is equivalent to the following minimization problem

$$\min_{W, W^T\mathbf{1}=\mathbf{1}} \sum_{i=1}^{d}\sum_{j=1}^{n}\left(w_{ij}e_{ij}^2 + \frac{\lambda_1}{2}w_{ij}^2\right) \tag{11}$$

and it is equivalent to the following problem

$$\min_{W, W^T\mathbf{1}=\mathbf{1}} \sum_{i=1}^{d}\sum_{j=1}^{n}\left(w_{ij} + \frac{e_{ij}^2}{\lambda_1}\right)^2 \tag{12}$$

Problem (12) is independent for different j, so we can optimize the following problem separately for each column w_j of W

$$\min_{w_j, w_j^T\mathbf{1}=\mathbf{1}} \sum_{j=1}^{n} \|w_j + \frac{1}{\lambda_1}h_j\|_2^2 \tag{13}$$

where h_j is the jth column of matrix $H = E \odot E$.

According to [29], problem (13) can be transformed into the following Lagrangian function

$$L(h_j, \eta, \beta_j) = \frac{1}{2}\|w_j + \frac{1}{\lambda_1}h_j\|_2^2 - \eta_j\left(w_j^T\mathbf{1} - 1\right) - \beta_j^T w_j \tag{14}$$

where η_j and $\beta_j \geq 0$ are the Lagrangian multipliers.

The optimal solution w_j should satisfy that the derivative of (14) with respect to w_j is equal to zero

$$\frac{\partial L}{\partial w_j} = w_j + \frac{h_j}{\lambda_1} - \eta_j\mathbf{1} - \beta_j = 0 \tag{15}$$

For the ith element of w_j, we have

$$w_{ij} + \frac{h_{ij}}{\lambda_1} - \eta_j\mathbf{1} - \beta_{ij} = 0 \tag{16}$$

By combining (16) and the Karush-Kuhn-Tucker(KKT) condition $w_{ij}\beta_{ij} = 0$ [29], we will have

$$w_j = \left(\eta_j\mathbf{1} - \frac{h_{ij}}{\lambda_1}\right)_+ \tag{17}$$

where $(v)_+ = max(0, v)$, according to (17) and the constraint $w_j^T\mathbf{1} = 1$, we have

$$\begin{aligned} &\sum_{i=1}^{d}\left(\eta_j - \frac{h_{ij}}{\lambda_1}\right) = 1 \\ \Rightarrow \quad &\eta_j = \frac{1}{d} + \frac{1}{d\lambda_1}\sum_{i=1}^{d} h_{ij} \end{aligned} \tag{18}$$

Therefore, we can obtain the optimal solution w_j according to (17).

Update E: for updating E, we have the following minimization problem

$$\min_E \|W^{\frac{1}{2}} \odot E\|_F^2 + \frac{\mu}{2}\|X - XZ - E - \frac{Y_1}{\mu}\|_F^2 \tag{19}$$

which can be further simplified to

$$\min_E \|W^{\frac{1}{2}} \odot E\|_F^2 + \frac{\mu}{2}\|E - S\|_F^2 \tag{20}$$

where $S = X - XZ - \frac{Y_1}{\mu}$. By spanning the Frobenius norm and removing the irrelevant terms, we have

$$\begin{aligned} &\min_E \sum_{i=1}^{d}\sum_{j=1}^{n}\left(w_{ij}e_{ij}^2 + \frac{\mu}{2}(e_{ij} - s_{ij})^2\right) \\ \Leftrightarrow &\sum_{i=1}^{d}\sum_{j=1}^{n}\min_{e_{ij}}\left(e_{ij} - \frac{\mu s_{ij}}{\mu + 2w_{ij}}\right)^2 \end{aligned} \tag{21}$$

The optimal solution to each element e_{ij} of variable E is

$$e_{ij} = \frac{\mu s_{ij}}{\mu + 2w_{ij}} \tag{22}$$

Update U: for updating U, problem (9) is transformed as follows

$$\min_{U} \lambda_2 \|U\|_* + \frac{\mu}{2}\|Z - U - \frac{Y_2}{\mu}\|_F^2 \tag{23}$$

This problem has a closed-form solution by using the singular value thresholding (SVT) operator [21], i.e.

$$U = \Theta_{\frac{\lambda_2}{\mu}}\left(Z + \frac{Y_2}{\mu}\right) = U\mathcal{S}_{\frac{\lambda_2}{\mu}}(\Sigma)V^T \tag{24}$$

where $U\Sigma V^T$ is the singular value decomposition of $\left(Z + \frac{Y_2}{\mu}\right)$, and $\mathcal{S}_{\frac{\lambda_2}{\mu}}(\cdot)$ is the soft-thresholding operator [21].

Update Z: when the other variables are fixed, the objective optimization problem (9) with respect to Z is degenerated to the following problem

$$\begin{aligned}&\min_{Z} \lambda_3 Tr(D^T Z) + \lambda_4 Tr(Z^T L_e Z)\\ &+\frac{\mu}{2}\left(\|X - XZ - E - \frac{Y_1}{\mu}\|_F^2 + \|Z - U - \frac{Y_2}{\mu}\|_F^2\right)\\ &\text{s.t.}\quad Z \geqslant 0, diag(Z) = 0, Z\bar{\mathbf{1}} = \bar{\mathbf{1}}\end{aligned} \tag{25}$$

where $\bar{1}$ is the column vector with all elements except the ith element are one, and the ith element is zero. We first calculate a latent solution $\hat{Z}$ by solving the following problem

$$\begin{aligned}&\min_{Z} \lambda_3 Tr(D^T Z) + \lambda_4 Tr(Z^T L_e Z)\\ &+\frac{\mu}{2}\left(\|X - XZ - E - \frac{Y_1}{\mu}\|_F^2 + \|Z - U - \frac{Y_2}{\mu}\|_F^2\right)\end{aligned} \tag{26}$$

This problem (26) has a closed-form solution as

$$\hat{Z} = (\lambda_4 L_e + X^T X + I)^{-1}\left(X^T\left(X - E + \frac{Y_1}{\mu}\right) + U - \frac{Y_2}{\mu} - \frac{\lambda_3}{\mu}D\right) \tag{27}$$

The optimal solution Z can be calculated by minimizing the following problem

$$\min_{Z \geqslant 0, diag(Z)=0, Z\bar{\mathbf{1}}=\bar{\mathbf{1}}} \|Z - \hat{Z}\|_F^2 \tag{28}$$

Similar to the optimization strategy of problem (13), we obtain each row of Z by

$$z_i = (\xi_i \bar{1}^T + \bar{z}_i)_+ \tag{29}$$

where $\bar{z}_i = [\bar{z}_{i1}, ..., \bar{z}_{ii}, ..., \bar{z}_{in}]$ is the ith row of $\hat{Z}$ (obtained by (27)), and the element $\bar{z}_{ii}$ is set to zero. Similar to problem (13), the Lagrangian multiplier ξ is

$$\xi_i = \frac{1 + \bar{z}_i \bar{\mathbf{1}}}{n - 1} \tag{30}$$

From (29), we can obtain the optimal solution z_i, which is the row of Z.

Algorithm 1 LSPLRG (Solving Problem (7) by ADMM).

```
Input: Data matrix $X \in \mathbb{R}^{d\times n}$, parameters $\lambda_1, \lambda_2, \lambda_3, \lambda_4 > 0$.
Initialization: $W = \mathbf{1}^T\mathbf{1}$, $Z = U = 0$, $E = X - XZ$, $Y_1 = Y_2 = 0$, $\mu = 0.01$,
$\mu_{max} = 10^8$, $\rho = 1.1$.
while not converged do
    1. Update variable $W$ as (17)
    2. Update variable $E$ as (22)
    3. Update variable $U$ as (24)
    4. Update variable $Z$ as (29)
    5. Update Lagrange multipliers $Y_1$ and $Y_2$
        $Y_1^{(k+1)} = Y_1^{(k)} + \mu^k(X - XZ^{(k+1)} - E^{(k+1)})$
        $Y_2^{(k+1)} = Y_2^{(k)} + \mu^k(Z^{(k+1)} - U^{(k+1)})$
    6. Update variable $\mu$:
        $\mu^{(k+1)} = \min(\mu_{max}, \rho\mu^{(k)})$
    7. Check the convergence conditions:
        $(\|X - XZ^{(k+1)} - E^{(k+1)}\|_F + \|Z^{(k+1)} - U^{(k+1)}\|_F)/\|X\|_F < tol$
end while
Output: an optimal solution $\{Z^*, W^*\}$
```

After we optimize variables W, E, U and Z, the ADMM algorithm also needs to update the Lagrange multipliers Y_1, Y_2 as well as parameter μ for faster convergence. The details of the optimization algorithm are exhibited in Algorithm 1.

Once obtaining the affinity graph Z^*, Normalized cut (Ncut) spectral clustering is applied on $(|Z^*| + |Z^{*T}|)/2$ to group data into several groups.

4 Experimental results

In this section, we consider the face clustering problem, where the goal is to group the face images into clusters according to their subjects. The clustering performance is evaluated on four publicly available image data sets, namely ORL,[1] Extended YaleB,[2] AR,[3] and LFW.[4] The important statistics of these databases are summarized in Table 1. To evaluate the performance of the proposed method, we conduct some experiments for face clustering, and compare the experimental results with several representative methods, including K-means, LRR [23], SSC[13], RLRR [9], BDR [24] and AWNLRR [36]. K-means serves as a baseline and obtains the final clustering results without learning an affinity matrix. The other six methods apply the same Ncut spectral clustering on the affinity matrix to obtain the clustering results. For fair comparison in all experiments, we use the Matlab codes released by the corresponding authors with the default or optimal parameter settings. The parameter

[1] http://www.cad.zju.edu.cn/home/dengcai/Data/FaceData.html.

[2] http://www.cad.zju.edu.cn/home/dengcai/Data/FaceData.html

[3] http://www2.ece.ohio-state.edu/~aleix/ARdatabase.html

[4] http://vis-www.cs.umass.edu/lfw/

Table 1 Description of databases

Dataset	Size(n)	Dimensionality(m)	# of classes
ORL	400	1024	40
Extended Yale B	2414	1024	38
AR	2600	19800	100
LFW	1251	1024	86

values of each method remain the same for all evaluation runs, and these values are set as suggested as in the corresponding literature.

4.1 Evaluation metrics

The clustering performance is evaluated by comparing the obtained label of each sample with that provided by the databases. Two metrics are used to quantitatively evaluate the clustering performance [5]. One metric is accuracy (AC) and the other is the normalized mutual information metric (NMI).

Given a data sample x_i, let r_i and s_i be the cluster label obtained by applying different algorithms and the label provided by the data set, respectively. The AC measures the percentage of correctly classified data points in the clustering solution compared with the ground truth class labels and is defined by

$$AC = \frac{\sum_{i=1}^{N} \delta(r_i, map(s_i))}{N} \tag{31}$$

where N is the total number of samples, and $\delta(x, y)$ is 1 if $x = y$ and 0 otherwise. $map(s_i)$ is a mapping function which can map the labels obtained by the clustering methods to the labels provided by the databases. The best mapping is usually fulfilled by the Kuhn-Munkres algorithm [8].

The second metric used in clustering applications is the normalized mutual information (NMI). It aims to measure the similarity of two clusters based on the amount of statistical information shared by random variables. Given two sets of image clusters $\mathcal{C} = \{c_1, ..., c_k\}$ and $\mathcal{C}' = \{c'_1, ..., c'_k\}$, their mutual information metric MI($\mathcal{C}, \mathcal{C}'$) is defined as:

$$MI(\mathcal{C}, \mathcal{C}') = \sum_{c_i \in \mathcal{C}, c'_j \in \mathcal{C}'} p(c_i, c'_j) \cdot log \frac{p(c_i, c'_j)}{p(c_i) \cdot (c'_j)} \tag{32}$$

where $p(c_i)$, $p(c'_j)$ represent the probabilities that an image arbitrarily selected from the data set belongs to the clusters $\mathcal{C}$, $\mathcal{C}'$, respectively, and $p(c_i, c'_j)$ represents the joint probability that this arbitrarily selected image belongs to both clusters simultaneously. MI($\mathcal{C}, \mathcal{C}'$) takes values between zero and max($H(\mathcal{C}), H(\mathcal{C}')$), where $H(\mathcal{C})$ and $H(\mathcal{C}')$ denote the entropy of the clusters $\mathcal{C}$ and $\mathcal{C}'$, respectively. MI($\mathcal{C}, \mathcal{C}'$) reaches the maximum when two sets of image clusters are identical, while it equals to zero when the two sets are completely independent. The advantage of MI($\mathcal{C}, \mathcal{C}'$) is that the value keeps the same for all kinds of permutations. By dividing the mutual information by max($H(\mathcal{C}), H(\mathcal{C}')$), NMI is derived as follows:

$$NMI(\mathcal{C}, \mathcal{C}') = \frac{MI(\mathcal{C}, \mathcal{C}')}{max(H(\mathcal{C}), H(\mathcal{C}'))} \tag{33}$$

Different from AC, NMI is invariant with the permutation of labels. Namely, it does not require the mapping between two clusters in advance.

Fig. 1 Sample face images from the ORL database

4.2 Clustering results

Given a collection of face images from multiple subjects, which have various illumination conditions and expressions. For each given clustering number k, the experiments are repeated 20 runs on the randomly chosen clusters and the average clustering performance is recorded as the final result. The accuracy (AC) and normalized mutual information (NMI) are calculated by the predicted and given labels, and the best results for each database are highlighted in bold in the tables.

4.2.1 ORL database

The ORL database has ten different images of each of 40 distinct subjects. For each subject, the images are captured under different facial expressions and light conditions. The face images used in this work are cropped and pre-resized to 32 × 32 pixels for computational efficiency. Images are preprocessed in advance so that faces are located. The samples of ORL database are depicted in Fig. 1. In order to evaluate the algorithm performances over different sample sizes, different cluster numbers are selected in our experiments. We select {10,15,20,25,30,35,40} clusters respectively from ORL database. Tables 2 and 3 show the detailed clustering accuracy and normalized mutual information by seven methods, respectively. Our proposed method outperforms all the other competing methods consistently, in terms of AC, while SSC performs slightly better as the $k = \{35, 40\}$ in terms of NMI. This can be explained by that the sparsity has the same ability to characterize the structure of data as that of low-rank.

Table 2 Clustering Performance on the ORL Database: AC (%)

k	K-means	LRR	SSC	RLRR	BDR	AWNLRR	LSPLRG
10	66.20	69.40	72.45	70.15	60.40	73.90	74.55
15	60.80	66.50	70.23	66.60	51.73	70.83	73.47
20	57.12	67.88	66.15	66.83	48.80	64.65	70.23
25	55.14	66.02	66.20	67.18	46.84	58.50	69.42
30	54.78	66.52	66.58	66.33	44.77	52.58	69.08
35	52.60	64.79	67.03	67.01	46.54	48.63	68.63
40	51.76	64.19	66.54	65.29	46.40	45.12	69.83
$Avg.$	56.92	66.47	67.88	67.06	49.35	59.17	70.74

Table 3 Clustering Performance on the ORL Database: NMI (%)

k	K-means	LRR	SSC	RLRR	BDR	AWNLRR	LSPLRG
10	73.16	73.87	79.28	73.56	66.16	78.88	80.59
15	71.39	74.90	80.00	75.19	61.49	79.02	81.90
20	70.34	77.71	78.97	76.77	59.99	75.01	81.22
25	70.93	77.53	80.21	78.02	59.14	70.69	81.64
30	70.53	78.70	81.39	78.74	61.36	65.63	81.72
35	70.76	78.28	82.70	79.80	65.40	64.12	81.69
40	70.85	78.88	82.84	79.47	66.56	62.90	82.82
$Avg.$	71.14	77.12	80.77	77.36	62.87	70.89	81.65

4.2.2 AR database

The AR face database contains about 4000 face images of 126 subjects. For each subject, there are 26 images are taken in two separate sessions with large variations in terms of facial disguise, illumination and expressions. Some images of the same subject from the AR face database are shown as in Fig. 2. Each data point is normalized to have a unit length. We then construct the data matrix X from subsets which consist of different numbers of subjects $k \in \{10, 12, 14, 16, 18, 20, 22, 24, 26\}$. The subspace clustering methods can be performed on X and the performances are recorded in the Tables 4 and 5. It can be seen LSPLRG achieves the competitive performance in most cases. One may notice that BDR distinctly outperforms other methods in this database. This is because the relationship between the data is more important in this database, which can be effectively exploited by block diagonal representation. Thus, beyond the sparse vector, low-rank matrix, the block diagonal matrix is another interesting structure of structured sparsity.

4.2.3 LFW database

The third database is the LFW face database, which consists of more than 13000 face images from 1680 subjects pictured under the unconstrained conditions. In our experiments, we select a subset including 1251 face images of 86 individuals for evaluation, and each subject has only 10-20 images with an imbalanced number of samples. All the face images are cropped and resized to 32 × 32 pixels. Figure 3 shows typical face images from LFW face database. In our experiments, we select {10,20,30,40,50,60,70,80} clusters respectively and the experimental results of different methods on this database are presented in Table 6 and 7.

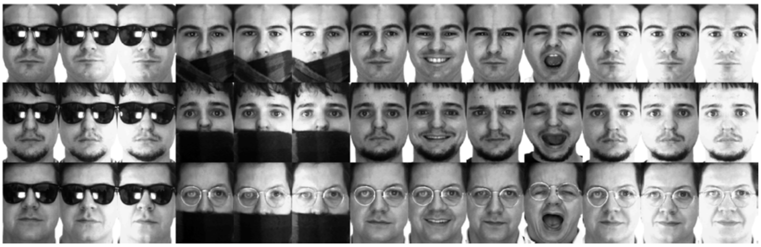


Fig. 2 Sample images from the AR database

Table 4 Clustering Performance on the AR Database: AC (%)

k	K-means	LRR	SSC	RLRR	BDR	AWNLRR	LSPLRG
10	26.88	46.62	18.55	45.77	52.81	48.29	50.38
12	24.15	48.48	17.44	49.33	50.38	48.24	49.74
14	21.59	50.59	16.02	50.69	52.25	48.64	50.58
16	20.42	49.05	15.26	47.14	54.62	49.86	52.01
18	19.79	48.70	14.29	47.97	54.34	49.42	52.87
20	18.90	49.14	13.42	48.29	56.46	49.28	52.86
22	17.95	50.59	13.22	50.37	58.72	49.96	52.36
24	18.41	50.43	12.39	50.66	60.32	48.81	52.28
26	17.11	51.66	12.07	51.20	58.48	48.99	52.52
Avg.	20.58	49.47	14.74	49.05	55.38	49.05	51.73

We can see that the best clustering results are still achieved by our LSPLRG, which also verifies the fact that the proposed method has particular potential for face image clustering.

4.2.4 Extended yale B database

The Extended Yale B face database contains 2414 frontal-face images of 38 subjects captured under different laboratory-controlled illumination conditions. For each subject, there are 59-64 nearly frontal images which are manually aligned and cropped. In our experiment, each image is resized to 32 × 32 pixels, and is vectorized to a 1024 vector as a data point. Figure 4 shows some face images with various lighting condition. Each data point is normalized to have a unit length. We construct the data matrix X from subsets which consist of different numbers of subjects $k \in \{10, 15, 20, 25, 30, 35\}$. It should be noted that Extended Yale B database is challenging for subspace clustering due to corruptions in the data caused by specular reflections. According to the Tables 8 and 9, we can see that the proposed LSPLRG once again achieves the best results on all cases. The average clustering accuracies obtained by LRR, SSC, RLRR, BDR, AWNLRR, LSPLRG are 76.79%, 73.53%, 61.25%,

Table 5 Clustering Performance on the AR Database: NMI (%)

k	K-means	LRR	SSC	RLRR	BDR	AWNLRR	LSPLRG
10	27.03	44.93	8.64	44.44	54.18	50.81	53.55
12	26.57	49.12	9.08	50.22	54.12	53.00	56.39
14	26.41	52.87	10.50	52.69	57.21	55.07	58.86
16	26.48	53.06	11.52	52.25	61.40	57.49	60.81
18	26.99	54.16	13.01	53.89	61.96	58.31	62.50
20	27.05	55.32	13.85	55.15	65.22	58.88	63.15
22	27.56	57.58	15.13	57.79	68.42	60.09	63.36
24	28.93	58.46	15.49	58.92	68.64	59.97	64.12
26	28.92	60.03	16.31	59.98	69.29	60.71	65.04
Avg.	27.33	53.95	12.61	53.93	62.27	57.15	60.86

Fig. 3 Sample images from the LFW database

Table 6 Clustering Performance on the LFW Database: AC (%)

k	K-means	LRR	SSC	RLRR	BDR	AWNLRR	LSPLRG
10	40.05	45.14	47.36	44.05	31.09	45.14	47.86
20	30.38	37.91	39.98	37.70	23.55	32.68	40.52
30	28.97	34.42	36.09	34.91	18.30	26.50	36.73
40	26.11	32.70	34.36	33.05	16.02	24.94	34.98
50	24.84	31.17	31.84	31.58	13.55	24.04	33.41
60	23.53	29.98	29.56	30.52	11.36	23.27	32.42
70	22.64	29.81	28.75	30.22	10.48	22.70	32.04
80	22.04	29.35	26.53	29.48	10.48	22.57	31.32
Avg.	27.32	33.81	34.31	33.94	16.85	27.73	36.16

Table 7 Clustering Performance on the LFW Database: NMI (%)

k	K-means	LRR	SSC	RLRR	BDR	AWNLRR	LSPLRG
10	38.76	45.62	48.10	44.74	26.83	45.96	49.34
20	40.71	49.35	51.44	49.65	26.67	42.40	52.71
30	43.86	52.07	53.44	52.40	24.55	39.17	54.22
40	44.63	53.56	55.21	54.16	23.23	42.71	55.64
50	45.66	54.69	55.39	54.83	21.17	45.70	56.64
60	46.57	55.37	54.95	55.85	18.69	47.80	57.54
70	47.53	56.40	55.52	56.80	18.14	48.85	58.55
80	48.34	57.36	55.33	57.61	19.28	50.33	58.99
Avg.	44.51	53.05	53.67	53.26	22.32	45.37	55.45

Fig. 4 Sample images from the Extended Yale B database

Table 8 Clustering Performance on the Extended Yale B Database: AC (%)

k	K-means	LRR	SSC	RLRR	BDR	AWNLRR	LSPLRG
10	18.16	81.50	74.01	83.37	44.40	94.04	95.19
15	14.55	77.55	73.55	66.36	40.09	91.88	93.78
20	13.12	76.31	69.33	54.79	41.41	90.18	92.70
25	12.02	74.23	71.30	52.25	48.16	88.63	90.47
30	11.16	75.04	75.16	54.36	52.06	87.50	89.43
35	10.99	76.13	77.85	56.36	55.77	86.51	88.97
Avg.	13.33	76.79	73.53	61.25	46.98	89.79	91.76

46.98%, 89.79%, and 91.76%, respectively. In this database, AWNLRR and LSPLRG significantly outperform the other methods in term of both accuracy and normalized mutual information. This is because there is a graph manifold regularization to preserve the local geometrical structure in these two methods. Moreover, the experimental results also validate that our method has an outstanding capability on overcoming the challenges of illumination variations and corruptions.

4.3 Visualization of the affinity matrices

To further demonstrate the effectiveness of LSPLRG, we evaluate the affinity matrices obtained by different methods, which could qualitatively reflect the performance of affinity learning. Figure 5 shows the visualization of these affinity matrices.

We use 10 classes of samples from ORL face database to visually present the affinity matrices derived by SSC, AWNLRR and LSPLRG. From Fig. 5, we can see that the affinity matrix designed by our method LSPLRG has more clear block-diagonal structure. Specifically, in the case of ORL, due to the existence of noise produced by face expressions and minor face poses, the main concern is the off-diagonal parts. Comparing the affinity matrices of SSC and AWNLRR, there is fewer nonzero entries in off-diagonal blocks obtained by LSPLRG and the entries within the diagonal blocks dominate the matrix in amplitude, which implies that each subject becomes highly compact and the different subjects are better separated. This demonstrates that LSPLRG not only take into account the global and local structures of data, but also learns a sparse affinity matrix. Thus, the affinity matrix learned by our method has more discriminative information and better for face clustering.

Table 9 Clustering Performance on the Extended Yale B Database: NMI (%)

k	K-means	LRR	SSC	RLRR	BDR	AWNLRR	LSPLRG
10	8.68	80.32	69.60	80.36	42.38	93.16	94.45
15	9.84	77.57	71.98	67.29	42.80	92.88	94.16
20	11.09	76.53	69.42	58.23	45.83	92.08	93.39
25	11.89	75.42	71.99	57.08	52.86	91.48	92.38
30	13.10	76.84	75.79	58.96	57.28	90.95	91.97
35	14.39	77.93	78.49	60.48	60.93	90.22	91.49
Avg.	11.50	77.44	72.88	63.73	50.35	91.80	92.97

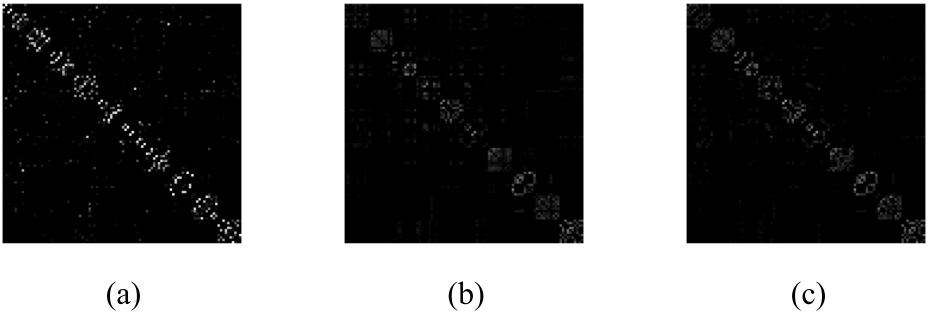


Fig. 5 Affinity matrix comparisons on the ORL database. (**a**)-(**c**) are the affinity matrices obtained using SSC, AWNLRR and LSPLRG, respectively

4.4 Convergence and parameter discussion

Convergence is the basic requirement for an excellent algorithm. In this subsection, we mainly focus on analyzing the convergence property of the proposed method with ADMM reported in Algorithm 1. In fact, the optimization scheme for problem (7) is equivalent to a two-block optimization problem, which is similar to the classical ADMM [16, 39]. The classical ADMM is intended to solve problems in the form

$$\min_{z\in\Omega_z, w\in\Omega_w} f(z)+h(w) \quad \text{s.t.} \quad Rz+Tw=u \tag{34}$$

where f and h are convex functions. Ω_z and Ω_w are the boundary constraints of variables z and w. It is apparent that ADMM for problem (34) can be directly extended to solve the matrix optimization problem as follows

$$\min_{Z\in\Omega_Z, W\in\Omega_W} f(Z)+h(W) \quad \text{s.t.} \quad RZ+TW=U \tag{35}$$

where R, T, U are matrices. The augmented Lagrangian of problem (35), in the method of classical ADMM, is formulated as

$$L(Z,W,C)=f(Z)+h(W)+\frac{\mu}{2}\|RZ+TW-U\|_F^2+\langle C, RZ+TW-U\rangle \tag{36}$$

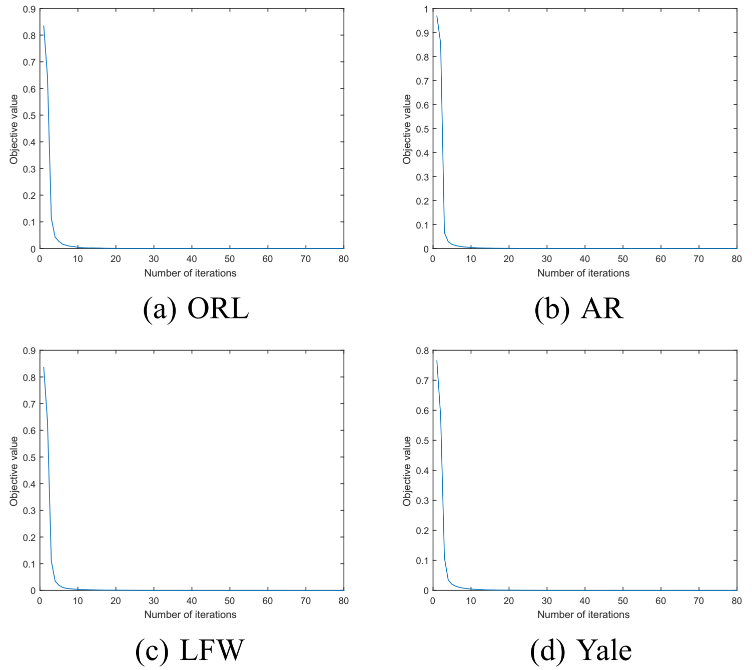
where C is the Lagrangian multiplier, and μ is a penalty coefficient. ADMM updates two primal variables in an alternating scheme, and iteratively solves problem (36) as follows

$$Z^{t+1}=\arg\min_Z L_\mu(Z,W^t,C^t) \tag{37}$$

$$W^{t+1}=\arg\min_W L_\mu(Z^t,W,C^t) \tag{38}$$

$$C^{t+1}=C^t+\mu(RZ^{t+1}+TW^{t+1}-U) \tag{39}$$

It should be noted that problem (7) is a special case of problem (35), and the proposed optimization algorithm shown in Algorithm 1 has same optimization style of classical ADMM. Therefore, the proposed optimization algorithm shown in Algorithm 1 is equivalent to a two-block ADMM, the global convergence of which is theoretically guaranteed [12, 16]. Figure 6 shows the convergence curves on four different databases, i.e. ORL, AR, LFW and Extended Yale B. It is obvious that the objective value decreases monotonously in each iteration, and finally converges to a local optima. Meanwhile, the optimization algorithm



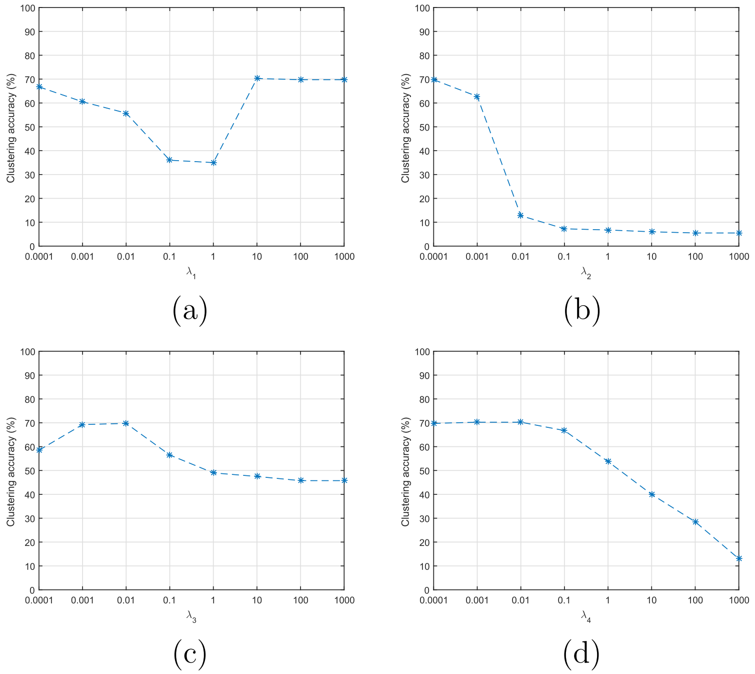
(a) ORL

(b) AR

(c) LFW

(d) Yale

Fig. 6 Convergence curves on different databases



(a)

(b)

(c)

(d)

Fig. 7 Clustering accuracy versus different values of parameters

converges fast and within ten iterations, which also proves the fast convergence property of the optimization.

There are several regularization parameters affecting the performance of our proposed method LSPLRG. λ_1 controls the values of weighted matrix W, which is used to avoid trivial solution. $\lambda_2, \lambda_3, \lambda_4$ are tunable parameters used to balance the importance of the corresponding terms. We investigate their impacts on the final clustering performance. In our experiments, we just change one parameter while fixing the other ones. Figure 7 shows the clustering accuracies of the proposed method with respect to the four parameters on the ORL database. Figure 7a shows the influence of the parameter λ_1 on the clustering accuracy of LSPLRG. Generally, larger λ_1 leads to better clustering performance. From Fig. 7b, we can see that the performance is not very well when λ_2 is larger than 0.001. The performance degeneration is due to the relatively weak regularization effect. From Fig. 7c, we can see that LSPLRG performs well and stably when $\lambda_3 \leq 0.01$. If λ_3 is relatively large, the local geometrical structure of data may not be well preserved due to the reconstruction loss. From Fig. 7d, it can be seen that the performance of LSPLRG is stable when λ_4 is within the range of $\{0.0001, 0.001, 0.01\}$. Based on these observations, we fix parameter $\lambda_1 = 100$, $\lambda_2 = 0.0001$, $\lambda_3 = 0.01$ and $\lambda_4 = 0.001$ for all the experiments in this paper.

5 Conclusion and future work

In this paper, a novel face clustering method called learning a sparsity preserving low-rank graph (LSPLRG) has been put forward. The proposed method jointly combines the sparse representation (SR) and low-rank representation (LRR) to learn the optimal affinity graph for clustering. In contrast with existing graph based methods, LSPLRG learns a initial affinity graph on the sparse coefficients without any a priori graph or similarity matrix. This is a critical step for reducing the influence of noise and outliers. Meanwhile, LSPLRG introduces an adaptive weighted matrix to constrain the self-representation, and integrates the distance regularization term into the low-rankness property of data representation to exploit the global and local structure of data. To reduce the redundant features, a constraint on the representation matrix to make our model learn a more discriminative graph for face clustering. Extensive experiments on benchmark databases show that the proposed method produce very competitive results for face clustering compared with several state-of-the-art subspace clustering methods.

Similar to most clustering methods, we found the main limitation of our method may be sensitive to some parameter initializations, which remains to our future study. Furthermore, we will try to develop our framework based on more effective features, and we plan to utilize the newly developed techniques including the block-level strategy [14] and the category-agnostic technique [45] to obtain the salient edge features. In order to reduce the redundant features and the influence of noise and outliers, the optimal affinity graph could be built by the salient edge features which integrates the local edge information and global location information.

Acknowledgments This work was supported by the National Natural Science Foundation of China under Grant no. 61572393, 71701021 and 41601437, the Basic Science Research of Shaanxi province under Grant no. 2018JQ1038, the Fundamental Research Funds for the Central Universities in Chang'an University under Grant no. 300102120201, and the Special Fund for Basic Scientific Research of Central Colleges in Chang'an University under Grant no. 310812163504.

References

1. Basri R, Jacobs DW (2003) Lambertian reflectance and linear subspaces. IEEE Trans Patt Anal Mach Intell 25(2):218–233
2. Benhur A, Horn D, Siegelmann H, Vapnik V (2002) Support vector clustering. J Mach Learn Res 2(2):125–137
3. Boyd S, Parikh N, Chu E, Peleato B, Eckstein J (2011) Distributed optimization and statistical learning via the alternating direction method of multipliers. Found Trends Mach Learn 3(1):1–122
4. Brukstein A, D Donoho M (2009) Elad, from sparse solutions of systems of equations to sparse modeling of signals and images. SIAM Rev 51(1):34–81
5. Cai D, He X, Han J (2005) Document clustering using locality preserving indexing. IEEE Trans Know Data Eng 17(12):1624–1637
6. Cai D, He X, Han J, Huang TS (2011) Graph regularized nonnegative matrix factorization for data representation. IEEE Trans Patt Anal Mach Intell 33(8):1548–1560
7. Candes EJ, Recht B (2009) Exact matrix completion via convex optimization. Found Comput Math 9(6):717–772
8. Chen S, Donoho DL, Saunders MA (2001) Atomic decomposition by basis pursuit. Siam Rev 43(1):129–159
9. Chen J, Yang J (2014) Robust subspace segmentation via low-rank representation. IEEE Trans Syst Man, Cybern 44(8):1432–1445
10. Donoho D (2006) For most large underdetermined systems of linear equations the minimal ℓ_1-norm solution is also the sparsest solution. Commun Pure Appl Math 59(6):797–829
11. Dornaika F, Kejani M, Bosaghzadeh A (2017) Graph construction using adaptive local hybrid coding scheme. Neural Netw 91–101
12. Eckstein J, Bertsekas DP (1992) On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. Math Program 55(3):293–318
13. Elhamifar E, Vidal R (2013) Sparse subspace clustering: algorithm, theory, and applications. IEEE Trans Pattern Anal Mach Intell 35(11):2765–2781
14. Fan D, Zhang S, Wu Y, Liu Y, Cheng M, Ren B, Rosin PL (2019) Scoot: a perceptual metric for facial sketches. Int Conf Comput Vision 5612–5622
15. Fang X, Xu Y, Li X, Lai Z, Wong WK (2016) Robust semi-supervised subspace clustering via non-negative low-rank representation. IEEE Trans on Syst Man, Cybern 46(8):1828–1838
16. Glowinski R, Tallec PL (1989) Augmented Lagrangian and operator-splitting methods in nonlinear mechanics. Math Comput 58(197)
17. Khan S, Hussain A, Usman M (2018) Reliable facial expression recognition for multi-scale images using weber local binary image based cosine transform features. Multimed Tools Appl 77(1):1133–1165
18. Khan S, Hussain A, Usman M, Nazir M, Riaz N, Mirza A (2014) Robust face recognition using computationally efficient features. J Intell Fuzzy Syst 27(6):3131–3143
19. Khan S, Ishtiaq M, Nazir M, Shaheen M (2018) Face recognition under varying expressions and illumination using particle swarm optimization. J Comput Sci 94–100
20. Khan S, Usman M, Riaz N (2015) Face recognition via optimized features fusion. J Intell Fuzzy Syst 28(4):1819–1828
21. Lin Z, Chen M, Ma Y (2011) The augmented Lagrange multiplier method for exact recovery of corrupted low-rank Matrices. Neural Inform Process Sys 1–20
22. Liu G, Li P (2016) Low-rank matrix completion in the presence of high coherence. IEEE Trans Signal Process 64(21):5623–5633
23. Liu G, Lin Z, Yan S, Sun J, Yu Y, Ma Y (2013) Robust recovery of subspace structures by low-rank representation. IEEE Trans Patt Analy Mach Intell 35(1):171–184
24. Lu C, Feng J, Lin Z, Mei T, Yan S (2019) Subspace clustering by block diagonal representation. IEEE Trans Pattern Anal Mach Intell 41(2):487–501
25. Luxburg UV (2007) A tutorial on spectral clustering. Stat Comput 17(4):395–416
26. Macqueen J (1967) Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth berkeleysymposium on mathematical statistics and probability, pp 281–297
27. Munir A, Hussain A, Khan S, Nadeem M, Arshid S (2018) Illumination invariant facial expression recognition using selected merged binary patterns for real world images. Optik 1016–1025
28. Ng A, Jordan M, Weiss Y (2002) On spectral clustering: analysis and an algorithm. Neural Inform Process Syst 849–856
29. Nie F, Wang X, Jordan M, Huang H (2016) The constrained laplacian rank algorithm for graph-based clustering. In: Proceedings of the thirtieth aaai conference on artificial intelligence, pp 1969–1976

30. Qiao L, Chen S, Tan X (2010) Sparsity preserving projections with applications to face recognition. Pattern Recognit 43(1):331–341
31. Rao S, Tron R, Vidal R, Ma Y (2010) Motion segmentation in the presence of outlying, incomplete, or corrupted trajectories. IEEE Trans Patt Anal Mach Intell 32(10):1832–1845
32. Shi J, Malik J (2000) Normalized cuts and image segmentation. IEEE Trans Pattern Anal Mach Intell 22(8):888–905
33. Vidal R (2011) Subspace clustering. IEEE Signal Process Mag 28(2):52–68
34. Wang Q, Lin J, Yuan Y (2016) Salient band selection for hyperspectral image classification via manifold ranking. IEEE Trans Neural Netw 27(6):1279–1289
35. Wen J, Han N, Fang X, Fei L, Yan K, Zhan S (2018) Low-rank preserving projection via graph regularized reconstruction. IEEE Trans on Syst Man, and Cybern 1–13
36. Wen J, Zhang B, Xu Y, Yang J, Han N (2018) Adaptive weighted nonnegative low-rank representation. Pattern Recognit 81:326–340
37. Xu L, Neufeld J, Larson B, Schuurmans D (2005) Maximum margin clustering. Neural Inform Process Syst 1537–1544
38. Xu R, Wunsch D (2005) Survey of clustering algorithms. IEEE Trans Neural Netw 16(3):645–678
39. Yang J, Yuan X (2012) Linearized augmented Lagrangian and alternating direction methods for nuclear norm minimization. Math Comput 82(281):301–329
40. Yao X, Han J, Zhang D, Nie F (2017) Revisiting co-saliency detection: a novel approach based on two-stage multi-view spectral rotation co-clustering. IEEE Trans Image Process 26(7):3196–3209
41. Yin M, Gao J, Lin Z (2016) Laplacian regularized low-rank representation and its applications. IEEE Trans Pat Anal Mach Intell 38(3):504–517
42. Zhang C, Hu Q, Fu H, Zhu P, Cao X (2017) Latent multi-view subspace clustering. Comput Vision Pattern Recognit 4333–4341
43. Zhang X, Xu C, Sun X, Baciu G (2016) Schatten-q regularizer constrained low rank subspace clustering model. Neurocomputing. 182:36–47
44. Zhao J, Hou Q, Ren B, Cheng M, Rosin P (2018) FLIC: fast linear iterative clustering with active search. Nat Conf Artif Intell 4(4):333–348
45. Zhao J, Liu J, Fan D, Cao Y, Yang J, Cheng M (2019) EGNEt: Edge guidance network for salient object detection. Int Conf Comput Vision 8779–8788
46. Zheng M, Bu J, Chen C, Wang C, Zhang L, Qiu G, Cai D (2011) Graph regularized sparse coding for image representation. IEEE Trans Image Process 20(5):1327–1336
47. Zheng J, Yang P, Chen S, Shen G, Wang W (2017) Iterative re-constrained group sparse face recognition with adaptive weights learning. IEEE Trans Image Process 26(5):2408–2423
48. Zhou T, Zhang C, Gong C, Bhaskar H, Yang J (2020) Multiview latent space learning with feature redundancy minimization. IEEE Trans on Syst Man, Cybern 50(4):1655–1668
49. Zhou T, Zhang C, Peng X, Bhaskar H, Yang J (2019) Dual shared-specific multiview subspace clustering. IEEE Trans Syst Man, Cybern 1–14