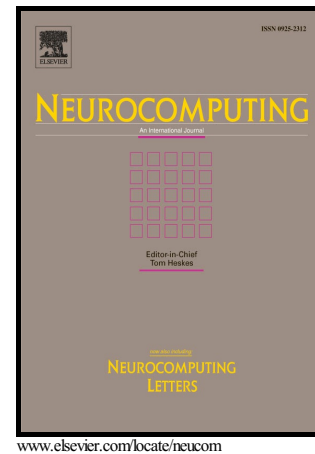


# Author's Accepted Manuscript

## Local and Global Regularized Sparse Coding for Data Representation

Zhenqiu Shu, Jun Zhou, Pu Huang, Xun Yu, Zhangjing Yang, Chunxia Zhao



PII: S0925-2312(15)01507-6  
DOI: <http://dx.doi.org/10.1016/j.neucom.2015.10.048>  
Reference: NEUCOM16226

To appear in: *Neurocomputing*

Received date: 26 May 2015  
Revised date: 26 September 2015  
Accepted date: 22 October 2015

Cite this article as: Zhenqiu Shu, Jun Zhou, Pu Huang, Xun Yu, Zhangjing Yang and Chunxia Zhao, Local and Global Regularized Sparse Coding for Data Representation, *Neurocomputing* <http://dx.doi.org/10.1016/j.neucom.2015.10.048>

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting galley proof before it is published in its final citable form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

# Local and Global Regularized Sparse Coding for Data Representation

Zhenqiu Shu<sup>1</sup>, Jun Zhou<sup>2</sup>, Pu Huang<sup>3</sup>, Xun Yu<sup>2</sup>, Zhangjing Yang<sup>4</sup>, Chunxia Zhao<sup>1</sup>

<sup>1</sup>School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China

<sup>2</sup>School of Information and Communication Technology, Griffith University, Nathan, QLD 4111, Australia

<sup>3</sup>School of Computer Science and Technology, Nanjing University of Posts and Telecommunications, Nanjing, 210023, China

<sup>4</sup>School of Technology, Nanjing Audit University  
Nanjing, 211815, China

**Abstract:** Recently, sparse coding has been widely adopted for data representation in real-world applications. In order to consider the geometric structure of data, we propose a novel method, *Local and Global regularized Sparse Coding (LGSC)*, for data representation. LGSC not only models the global geometric structure by a global regression regularizer, but also takes into account the manifold structure using a local regression regularizer. Compared with traditional sparse coding methods, the proposed method can preserve both global and local geometric structures of the original high-dimensional data in a new representation space. Experimental results on benchmark datasets show that the proposed method can improve the performance of clustering.

**Keywords:** sparse coding; data representation; regularizer; regression; clustering

## 1 Introduction

Over the past decade, data representation has attracted increasing attention in computer vision, information retrieval and machine learning. In many applications [1, 2, 3, 4], processing high dimensional data in classification or clustering tasks is still a big challenge. To improve the performance of classification or clustering, a common way is to seek a meaningful low dimensional representation of the high dimensional data by dimensionality reduction or matrix factorization approaches.

Principal component analysis (PCA) [5] and linear discriminant analysis (LDA) [6] are two widely used linear representation methods. The former is an unsupervised learning approach, which aims to preserve the global covariance structure of data. The latter is a

supervised learning method, which extracts the optimal discriminant vectors when class labels of data are available. However, both of them can not discover the latent manifold structure among data. In the past a few years, many methods have been proposed to address this issue. Among them, the most representative methods are ISOMAP [7], locally linear embedding (LLE) [8] and Laplacian Eigenmaps (LE) [9]. Although these manifold learning methods have achieved impressive results on data visualization, they cannot deal with the ‘out-of-sample’ problem. He et al [10] proposed a linear version of the LE, namely locality preserving projection (LLP), which can alleviate this drawback. Several data representation methods, such as local and global regressive mapping (LGRM) [11], graph regularized nonnegative matrix factorization (GNMF) [12], local learning regularized nonnegative matrix factorization (LLRNMF) [13], locally consistent concept factorization (LCCF) [14] and local regularized concept factorization (LCF) [15], have been developed to exploit the geometric manifold structure of data. Extensive experimental results have demonstrated the effectiveness of these techniques.

In recent years, sparse coding (SC) has shown great success in data representation and a range of applications such as image processing [16, 17, 18], classification [19, 20, 21, 22] and visual analysis [23, 24, 25, 26]. Essentially, SC seeks to linearly represent a test sample by only a few training samples, which leads to the sparsity of the representation coefficient. To achieve sparse representation, many methods have been developed in the past few years, e.g. sparse PCA [27], sparse NMF [28] and sparse low-rank representation [29]. However, in conventional sparse coding methods, a common drawback is that some prior knowledge of data has been neglected, such as the geometric structure information. Wang et al [30] presented a novel sparse coding method, called locality-constrained linear coding (LLC). Furthermore, in order to preserve the spatial consistency, locally-invariant sparse representation were proposed by pooling the sparse coefficients across overlapping windows [31]. Mairal et al [32] introduced a simultaneous sparse coding method by jointly decomposing groups of similar signals on subsets of the learned dictionary, which was implemented using a group-sparsity regularizer. Zheng et al [33] proposed a graph regularized sparse coding (GSC) method for image representation. In GSC, the geometric manifold structure of data is taken into account by imposing the graph regularizer. Thus, GSC performs significantly better than the traditional sparse coding methods on several benchmark databases. However, GSC only utilizes the local manifold structure of data by the regularization technique, and neglects the global geometric relationship of data. Therefore, a

better approach is expected to learn a lower-dimensional representation to preserve both local and global structure of data, which is beneficial for achieving promising performance.

Motivated by the recent progresses in sparse coding and manifold learning, in this paper, a novel method, *local and global regularized sparse coding (LGSC)*, is proposed to represent the high dimensional data. Compared with traditional sparse coding methods, the proposed LGSC not only considers the manifold structure of data by constructing a local regression predictor, but also preserves its global structure using a global regression regularizer. Experimental results on several bench mark datasets have validated the proposed the effectiveness of the LGSC methods.

It is worthwhile to highlight the main contributions of this work as follows:

- (1) We employ the local regression to model the local manifold structure, and simultaneously use the global regression as a regularization term to capture the global structure of data. In LGSC, both local and global regression regularization terms are combined into an integrated regularizer, which captures the intrinsic geometric structure of real-world data;
- (2) In LGSC, the integrated regularizer is incorporated into the traditional sparse coding method, which makes LGSC more discriminative. In addition, we develop an iterative update scheme to solve the optimization problem of the LGSC and present the convergence curve in this paper;
- (3) We conduct comprehensive experiments to analyse and compare our method with several state-of-the-art methods. The experimental results on real world image datasets demonstrate that the proposed method is superior to other data representation methods.

The rest of this paper is organized as follows. The sparse coding and GSC methods are reviewed in Section II. The proposed LGSC method is described in Section III. The optimization scheme of LGSC is introduced in Section IV. Experimental results are presented in Section V. Finally, conclusions are drawn in Section VI.

## 2 Related works

This section contains description of related works to the proposed approach, i.e. SC and GSC.

### 2.1 Sparse coding

Sparse coding aims to linearly represent a sample by a few atoms in a dictionary. Given a data set  $X = [x_1, x_2, \dots, x_n] \in R^{m \times n}$  with  $n$  data points sampled from an  $m$ -dimensional feature space. Let  $D \in R^{m \times k}$  be an over-complete dictionary and  $A \in R^{k \times n}$  be the representation coefficient, where  $k$  denotes the number of the atoms. In order to achieve the sparsity of

coding coefficients, the  $l_0$ -norm is used to constrain the representation coefficient. Mathematically, the minimization problem of sparse coding can be formulated as

$$\begin{aligned} \min_{D,A} \|X - DA\|_F^2 + \alpha \sum_{i=1}^m \|a_i\|_0 \\ \text{s.t. } \|d_i\|^2 \leq c, i = 1, \dots, k \end{aligned} \quad (1)$$

where  $\|\cdot\|_F$  and  $\|\cdot\|_0$  denote the Frobenius norm of a matrix and the  $l_0$ -norm of a vector, respectively,  $c$  is a given constant and  $\alpha$  is a constant parameter. Solving the  $l_0$ -norm minimization problem is NP-hard. Fortunately, it can be replaced by an  $l_1$ -norm minimization problem if the solution of Eq. (1) is sufficiently sparse [34, 35]. Therefore, the optimization problem in Eq. (1) can be rewritten as follows:

$$\begin{aligned} \min_{D,A} \|X - DA\|_F^2 + \alpha \sum_{i=1}^m \|a_i\|_1 \\ \text{s.t. } \|d_i\|^2 \leq c, i = 1, \dots, k \end{aligned} \quad (2)$$

where  $\|\cdot\|_1$  denotes the  $l_1$ -norm of a vector. Since the  $l_1$ -norm minimization problem in Eq. (2) is a convex optimization problem, it can be efficiently solved using existing software packages such as  $l_1$ -magic [36], PDCO-LSQR [37] and PDCO-CHOL[37].

## 2.2 Graph Regularized Sparse Coding

Previous studies [7, 8, 9] have shown that manifold learning plays an important role in data representation. A natural assumption is that if two data samples are close in the original feature space, then their low dimensional representation should be close to each other in the new representation space. This is usually referred to as the *manifold learning assumption*. Using graph regularization techniques, GSC can discover the latent manifold structure of data.

Given a set of data points  $X = [x_1, x_2, \dots, x_n] \in R^{m \times n}$ , the geometric structure of data can be characterized by a  $k$ -nearest neighbour graph  $G = \{X, W\}$  with a vertex set  $X$  and an affinity weight matrix  $W$ . If  $x_i$  is among the  $k$ -nearest neighbours of  $x_j$  or  $x_j$  is among the  $k$ -nearest neighbours of  $x_i$ ,  $W_{ij}=1$ , otherwise,  $W_{ij}=0$ . The graph regularization term is expressed as follows:

$$\frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m (a_i - a_j) W_{ij} = \text{Tr}(ALA^T) \quad (3)$$

where  $A = [a_1, \dots, a_n]$  is the sparse coefficient matrix,  $L = D - W$  is the Laplacian matrix,  $D$  is a diagonal matrix and  $D_{ii} = \sum_j W_{ij}$ .

By incorporating the Laplacian regularizer (3) into sparse coding, the objective function of GSC can be expressed as follows:

$$\begin{aligned} \min_{D,A} & \|X - SA\|_F^2 + \alpha \text{Tr}(ALA^T) + \beta \sum_{i=1}^m \|a_i\|_1 \\ \text{s.t.} & \|a_i\|^2 \leq c, i = 1, \dots, k \end{aligned} \quad (4)$$

where  $\alpha$  and  $\beta$  are the regularization parameters. The optimization problem in Eq. (4) can be solved by the feature search algorithm proposed in [38].

### 3 The proposed method

We start this section by discussing the motivation of our work. Then we introduce the proposed LGSC method in detail.

#### 3.1 Motivation

Sparse coding is a typical data representation method based on an over-complete dictionary. Most of sparse coding methods, however, fail to make full use of the geometrical structure of data. In fact, the intrinsic structure of data is unknown and complex in many real-world applications. Thus, a single global or local graph may be insufficient to characterize the underlying geometrical structure of data. A reasonable approach should integrate both local and global structures of data in the representation step.

Previous studies [7, 8, 9, 10, 12] have shown that modelling the local geometrical structure has achieved great success for data representation. Therefore, we employ the local regression regularization term to discover the latent geometric structure among data. Specifically, the complete data space is divided into a multitude of local regions. Each sample can be well predicted from the local region it belongs to. This is usually called *local learning* assumption [39]. One drawback of pure local learning, however, is the lack of data points in each local region for constructing a local classifier. To deal with this issue, we further exploit a kernelized global regression method to capture the global geometric structure of data. According to reference [11], we introduce a local and global regularization term to capture the intrinsic geometric structure.

In the following subsection, details on the local and global regularization are described.

#### 3.2 Local and Global Regularization

Given a data set  $X = [x_1, x_2, \dots, x_n] \in R^{m \times n}$ , where  $x_i \in R^{m \times 1}$  denotes a sample vector. Local learning assumption requires each sample be linearly represented by its neighbourhood. Let

$N(x_i)$  denote the  $k$ -nearest neighbours of  $x_i$ , the linear regression function  $f_i$  can be expressed as:

$$f_i(x) = W_i^T x_j + b_i \quad (5)$$

where  $x_j \in N(x_i)$ ,  $W_i$  and  $b_i$  represent the weight vector and bias of  $f_i$ , respectively. As a result, the lost function for each data point can be represented as follows:

$$J_i^{local} = \sum_{x_j \in N(x_i)} \|W_i^T x_j + b_i - a_j\|^2 + \gamma \|W_i\|^2 \quad (6)$$

where  $\gamma$  is a nonnegative parameter,  $a_i$  denotes the low dimensional representation of  $x_i$ . In this equation, the regularization term  $\gamma \|W_i\|^2$  measures the smoothness of  $W_i$ . Therefore, we can compute the sum of the prediction errors of all data as follows:

$$J^{local} = \sum_{i=1}^n \sum_{x_j \in N(x_i)} \|W_i^T x_j + b_i - a_j\|^2 + \gamma \|W_i\|^2 \quad (7)$$

This equation is called *Local Learning Regularization*.

To capture the global geometric structure of data, we employ kernel regression to represent each sample. The global empirical loss function  $J^{global}$  can be expressed as follows:

$$J^{global} = \sum_{i=1}^n \|\phi(W)^T \phi(x_i) + b - a_i\|^2 + \gamma \|\phi(W)\|_F^2 \quad (8)$$

where  $\phi(\cdot)$  denotes a kernel mapping function,  $b$  is a bias term. Here, Eq. (8) is called *Global Learning Regularization*. Combining Eq. (7) and Eq. (8), the local and global regularization term can be written as

$$\begin{aligned} J &= L^{local} + \mu L^{global} \\ &= \sum_{i=1}^n \sum_{x_j \in N_i} \left( \|W_i^T x_j + b_i - a_j\|^2 + \gamma \|W_i\|^2 \right) \\ &\quad + \mu \left[ \sum_{i=1}^n \|\phi(W)^T \phi(x_i) + b - a_i\|^2 + \gamma \|\phi(W)\|_F^2 \right] \end{aligned} \quad (9)$$

where  $\mu$  is used to balance the importance of the local and global structure of data. Let  $X_i = [x_i, x_{i1}, x_{i2}, \dots, x_{ik-1}] \in R^{m \times k}$  be the data matrix of  $N_i$  and  $A_i = [a_i, a_{i1}, \dots, a_{ik-1}]^T \in R^{m \times k}$  be the new representation of  $N_i$ , respectively. The model in Eq. (9) can be rewritten as follows:

$$J = \sum_{i=1}^n (\|X_i^T W_i + 1_k b_i^T - A_i\|^2 + \gamma \|W_i\|^2) + \mu (\|\phi(X)^T \phi(W) + 1_n b - A\|_F^2 + \gamma \|\phi(W)\|_F^2) \quad (10)$$

Where  $1_k \in R^k$  and  $1_n \in R^n$  are two vectors of all ones. Using the property that  $\|M\|_F^2 = \text{tr}(M^T M)$  for any matrix  $M$ , the local regularization term in Eq. (10) can be rewritten:

$$J^{local} = \sum_{i=1}^n \{ \text{tr}[(X_i^T W_i + 1_k b_i^T - A_i)^T (X_i^T W_i + 1_k b_i^T - A_i)] + \gamma \text{tr}(W_i^T W_i) \} \quad (11)$$

By taking the partial derivative of Eq. (11) with respect to  $W_i$  and  $b_i$ , we have

$$\frac{\partial J^{local}}{\partial W_i} = W_i^T X_i 1_k + k b_i - A_i^T 1_k \quad (12)$$

$$\frac{\partial J^{local}}{\partial b_i} = X_i X_i^T W_i + X_i 1_k b_i^T - X_i A_i + \gamma W_i \quad (13)$$

Let  $\frac{\partial J^{local}}{\partial W_i} = 0$  and  $\frac{\partial J^{local}}{\partial b_i} = 0$ , then

$$b_i = \frac{1}{k} (A_i^T 1_k - W_i^T X_i 1_k) \quad (14)$$

$$W_i = (X_i H_k X_i^T + \gamma I)^{-1} X_i H_k A_i \quad (15)$$

where  $H_k = I - \frac{1}{k} 1_k 1_k^T$  denotes the local centering matrix. Substituting Eq. (14) and Eq. (15) into Eq. (6), we get

$$\sum_{i=1}^n \text{tr}(A_i^T F_i A_i) \quad (16)$$

where  $F_i = H_k - H_k X_i^T (X_i H_k X_i^T + \gamma I)^{-1} X_i H_k$ . Define a selection matrix  $Q$  in which  $Q_{ij}=1$  if  $x_i$  is the  $j$ -th element in  $N_i$ , and  $Q_{ij}=0$  otherwise. It is easy to see that  $A_i = Q_i^T A$  and thus Eq. (16) can be reformulated as:

$$\sum_{i=1}^n \text{tr}(A_i^T Q_i F_i Q_i^T A_i) = \text{tr}(A^T (\sum_{i=1}^n Q_i F_i Q_i^T) A) \quad (17)$$

Meanwhile, the local learning regularization in Eq. (7) can be reformulated as



$$J^{local} = A^T L^{local} A \quad (18)$$

where  $L^{local} = \sum_{i=1}^n Q_i F_i Q_i^T$ .

Similarly, the second term in Eq. (9) can be written as

$$J_{global} = tr \left\{ \left[ \phi(X)^T \phi(W) + 1_n b^T - A \right]^T \left[ \phi(X)^T \phi(W) + 1_n b^T - A \right] \right\} + \gamma tr \left[ \phi(X)^T \phi(X) \right] \quad (19)$$

By taking the partial derivative of Eq. (11) with respect to  $W_i$  and  $b_i$  and let it be zero, we have

$$\begin{aligned} \phi(W) &= (\phi(X)H\phi(X)^T + \gamma I)^{-1} \phi(X)A \\ &= \phi(X)H(H\phi(X)^T \phi(X)H + \gamma I)^{-1} A \end{aligned} \quad (20)$$

$$\begin{aligned} b &= \frac{1}{n} A^T 1_n - \frac{1}{n} W^T \phi(X) 1_n \\ &= \frac{1}{n} A^T 1_n - \frac{1}{n} A^T (H\phi(X)^T \phi(X)H + \gamma I)^{-1} H\phi(X)^T \phi(X) 1_n \end{aligned} \quad (21)$$

Let  $H = I - \frac{1}{n} 1_n 1_n^T$  be the global centering matrix. Similarly, the global regularization term is rewritten as

$$J^{global} = A^T L^{global} A \quad (22)$$

Thus, we have

$$\begin{aligned} L^{global} &= H - H\phi(X)^T \left[ \phi(X)H\phi(X)^T + \gamma I \right]^{-1} \phi(X)H \\ &= \gamma H(H\phi(X)^T \phi(X)H + \gamma I)^{-1} H \end{aligned} \quad (23)$$

where  $\phi(X)^T \phi(X)$  can be calculated by a kernel function. Suppose the dot product of  $x_i$  and  $x_j$  in  $H$  is given by the following kernel function:

$$K_{x_i, x_j} = (\phi(x_i)\phi(x_j)) = \phi(x_i)^T \phi(x_j) \quad (24)$$

where  $K$  needs to satisfy Mercer's condition. Therefore,  $L^{global}$  can be computed by

$$L^{global} = \gamma H(HKH + \gamma I)^{-1} H \quad (25)$$

where  $K$  denotes the kernel matrix with its elements  $K_{ij} = K_{x_i, x_j}$ . Combining the local regularization term (18) and the global regularization term (25), the local and global regularization term (10) can be reformulated as follows:

$$\begin{aligned} L^{local-global} &= L^{local} + \mu L^{global} \\ &= A^T (L^{local} + \mu L^{global}) A \\ &= A^T \left( \sum_{i=1}^n Q_i F_i Q_i^T + \mu \gamma H (H K H + \gamma I)^{-1} H \right) A \end{aligned} \quad (26)$$

The Eq. (26) is called *local and global regularization*.

### 3.2 Objective function of LGSC

To incorporate the local and global structure of data into sparse coding, the proposed LGSC is applied for data representation by imposing the local and global regularization term in Eq. (26). Thus, the objective function of LGSC can be defined as follows:

$$\begin{aligned} \min_{S, A} & \|X - SA\|_F^2 + \alpha \text{Tr}(A L^{local-global} A^T) + \beta \sum_{i=1}^m \|a_i\|_1 \\ \text{s.t.} & \|a_i\|^2 \leq c, i = 1, \dots, k \end{aligned} \quad (27)$$

Clearly, the objective function of LGSC is similar to GSC. Thus, we can adopt the same solution scheme to optimize the model in Eq. (27). Obviously, the objective function of LGSC is nonconvex with respect to the product of  $S$  and  $A$ . Fortunately, the dictionary  $S$  and its corresponding coefficient  $A$  can be updated alternately in the optimization process. In the next subsection, we will introduce the optimization method of the proposed LGSC in detail.

### 3.3 Optimization

#### 3.3.1 Updating the dictionary $S$

In this subsection, we describe a method to learn the dictionary  $S$ . Fixing the coefficient matrix  $A$ , the optimization problem in Eq. (27) can be transformed into the following least squares problem with quadratic constraints:

$$\begin{aligned} \min_{D, A} & \|X - SA\|_F^2 \\ \text{s.t.} & \|a_i\|^2 \leq c, i = 1, \dots, k \end{aligned} \quad (28)$$

Obviously, Eq. (28) can be solved efficiently by calculating the Lagrangian Dual. Let  $\lambda = [\lambda_1, \lambda_2, \dots, \lambda_k]$  be the Lagrange multiplier vector, where  $\lambda_i$  is the Lagrange multiplier

associated with the  $i$ -th inequality  $\|a_i\|^2 \leq c$ . Therefore, the solution of Eq. (28) can be derived as:

$$S^* = XA^T(AA^T + \text{diag}(\lambda^*))^{-1} \quad (29)$$

where  $\lambda^*$  denotes the optimal solution of  $\lambda$ . For more details of this optimization method, please refer to [34].

### 3.3.2 Learning the coding coefficient $A$

By fixing  $S$ , we transform the optimization problem in Eq. (27) into the following problem:

$$\min_{S,A} \|X - SA\|_F^2 + \alpha \text{Tr}(AL^{local-global}A^T) + \beta \sum_{i=1}^m \|a_i\|_1 \quad (30)$$

Eq. (30) can be solved by a coordinate optimization method, which optimizes the coding coefficient one by one. In other words, we can optimize the  $i$ -th coefficient  $a_i$  in  $A$ , while fixing the remaining. Thus, we can solve the following optimization problem instead of Eq. (30):

$$\min_{a_i} \|X_i - Sa_i\|_F^2 + \alpha \left[ L_{ii}^{local-global} a_i^T a_i + 2a_i^T \sum_{j \neq i} L_{ij}^{local-global} a_j \right] + \beta \|a_i\|_1 \quad (31)$$

Similar to GSC, the optimization problem in Eq. (31) can be solved by the feature-sign search algorithm proposed in [33, 38].

### 3.4. The proposed LGSC algorithm

According to the above description, the algorithm of the proposed LGSC is summarized as follows.

---

#### Algorithm 1: Local and Global regularized Sparse Coding

---

Input: A data set of  $m$  data points  $X = [x_1, x_2, \dots, x_m]$ , iteration number  $T$ , parameters  $\alpha, \beta, \mu$  and  $\gamma$ .

Step 1: Calculate the local Laplacian matrix  $L^{local}$  and global Laplacian matrix  $L^{global}$  in Eq.(18) and Eq.(26), respectively;

Step 2: Derive the local-global Laplacian matrix  $L^{local-global}$  in Eq.(26);

Step 3: For  $i=1, \dots, T$  do

Step 4: Update the dictionary  $S$  in Eq. (29);

Step 5: Update the coding coefficient  $A$  using feature-sign search algorithm to solve the

optimization problem in Eq. (31);

Step 6: End for

Output: dictionary  $S$  and corresponding coefficient  $A$ .

---

#### 4 Experimental results

In this section, we present some experiments on three data sets (ORL, Isolet and USPS) to demonstrate the effectiveness of the proposed method. Note that the proposed LGSC method is an unsupervised learning method for which the label information of data is unavailable. Thus, clustering experiments are naturally carried out to validate the effectiveness of the proposed method. In the experiments, the accuracy and normalized mutual information are used to measure the quantitative performance of different data representation methods. To test the validity of the proposed LGSC method, we compare the performance of LGSC with some other methods, such as K-means, PCA, Ncut, SC, Sparse Concept Coding (SCC) [21] and GSC.

##### 4.1 Evaluation criteria

In this paper, accuracy (AC) and normalized mutual information (NMI) are used to measure the performance of different methods. Given a data point  $x_i$ , let  $r_i$  and  $s_i$  denote the cluster label calculated from the proposed algorithm and the label provided by the real dataset, respectively. The AC can be defined as follows:

$$AC = \frac{\sum_{i=1}^n \delta(s_i, \text{map}(r_i))}{n}$$

where  $n$  denotes the number of samples,  $\delta(x, y)$  is delta function and  $\text{map}(r_i)$  is the permutation mapping function that maps each cluster label  $r_i$  to the best label from the data set. We can use the Kuhn-Munkres algorithm [40] to seek the best mapping.

Suppose  $C$  and  $C'$  denote the set clusters in the dataset and obtained from the proposed algorithm, respectively. Mutual information (MI) is used to measure the similarity of two sets of data. The mutual information  $MI(C, C')$  is defined as follows:

$$MI(C, C') = \sum_{c_i \in C, c'_i \in C'} p(c_i, c'_i) \cdot \log_2 \frac{p(c_i, c'_i)}{p(c_i) \cdot p(c'_i)}$$

where  $p(c_i)$  and  $p(c'_i)$  are the probabilities that a data point is arbitrarily chosen from the database belongs to the clusters  $c_i$  and  $c'_i$ , respectively, and  $p(c_i, c'_i)$  denotes the joint

probability that a data point belongs to the clusters  $c_i$  and  $c'_i$  at the same time. Thus, the normalized mutual information  $\bar{MI}$  is defined as follows:

$$\bar{MI}(C, C') = \frac{MI(C, C')}{\max(H(C), H(C'))}$$

where  $H(C)$  and  $H(C')$  denotes the entropies of  $C$  and  $C'$ , respectively. Obviously, the value of  $\bar{MI}(C, C')$  ranges from 0 to 1. Specifically, if two sets of clusters are independent,  $\bar{MI}(C, C') = 0$ . If they are identical,  $\bar{MI}(C, C') = 1$ .

#### 4.2 ORL face database

The ORL face dataset includes 400 grayscale images of 40 different subjects. The images were taken at different times, with varying lighting condition, facial expressions and facial poses. All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position. In this dataset, the size of each face image is normalized to  $32 \times 32$ . Thus, each image can be resized to a 1024-dimensional vector. Some sample images from ORL face database are shown in Fig.1.



Fig.1 Sample images from the ORL database

In this experiment, we arbitrarily chose images in  $P(=30, 32, \dots, 40)$  categories to evaluate the proposed method. For each  $P$  value, we ran the experiment for 10 times and then list the average performance as the final result. For the proposed method, we use the Radial Basis Function as the kernel function whose standard deviation is set to 2. Table 1 shows the accuracy and normalized mutual information of all methods. Note that the average AC and NMI of GSC reach 59.9% and 74.6%, respectively. Meanwhile, the average AC and NMI of SCC are 54.9% and 72.4%, respectively. It can be observed that GSC and SCC perform better than SC on this dataset, as both GSC and SCC can discover the latent manifold structure of data. It is worth noting that GSC uses the manifold structure information of data as a regularizer. SCC, however, incorporates it into the basis vectors by spectral analysis. In addition, we can see that the proposed LGSC consistently outperforms the other methods on all configuration of cluster number  $P$ . The reason is that LGSC further captures the global structure information of data compared with GSC and SCC.

Table 1 The clustering performance of all methods on the ORL database

(a)AC							
$P$	K-means	PCA	Ncut	SC	SCC	GSC	LGSC
30	0.565	0.584	0.613	0.565	0.538	0.632	<b>0.652</b>
32	0.473	0.502	0.571	0.521	0.561	0.596	<b>0.611</b>
34	0.507	0.513	0.578	0.538	0.550	0.581	<b>0.607</b>
36	0.537	0.545	0.609	0.549	0.559	0.603	<b>0.615</b>
38	0.526	0.557	0.596	0.559	0.548	0.598	<b>0.608</b>
40	0.521	0.549	0.601	0.553	0.536	0.585	<b>0.609</b>
Avg	0.522	0.542	0.595	0.548	0.549	0.599	<b>0.617</b>
(b)NMI							
$P$	K-means	PCA	Ncut	SC	SCC	GSC	LGSC
30	0.724	0.737	0.752	0.713	0.711	0.761	<b>0.788</b>
32	0.674	0.697	0.731	0.699	0.7183	0.743	<b>0.749</b>
34	0.697	0.704	0.729	0.696	0.724	0.734	<b>0.755</b>
36	0.701	0.714	0.752	0.753	0.733	0.749	<b>0.772</b>
38	0.708	0.738	0.727	0.729	0.739	0.744	<b>0.763</b>
40	0.711	0.731	0.761	0.704	0.719	0.742	<b>0.764</b>
Avg	0.703	0.720	0.742	0.716	0.724	0.746	<b>0.765</b>

#### 4.3 Isolet spoken letter recognition database

Isolet spoken letter database contains 150 subjects who spoke the name of each letter of the alphabet twice. The speakers were grouped into sets of 30 people each, and the groups were named isolet1 through isolet5. In our experiment, we adopted isolet1 as the test data for the clustering task.

In Isolet1 dataset, we randomly selected  $P$  categories samples to evaluate all methods. Similarly, we also ran the experiments for 10 times on each  $P$  value for every method and recorded the average performance. For the proposed LGSC method, the Radial Basis Function is used as the kernel function and the kernel parameter is set to 2. The average results of all methods are shown in Table 2. It can be seen that the proposed LGSC method performs significantly better than the other methods in terms of accuracy or normalized mutual information. Specifically, SCC and GSC are superior to K-means, PCA, Ncut and SC. The reason is that both SCC and GSC consider the sparsity of data and the manifold geometric structure of data simultaneously. Compared with SCC and GSC, the proposed LGSC further utilizes the global structure information of data, and hence achieves the best result in the clustering task.

Table 2 The clustering performance of all methods on the Isolet database

(a)AC							
$P$	K-means	PCA	Ncut	SC	SCC	GSC	LGSC
16	0.693	0.716	0.703	0.561	0.684	0.698	<b>0.737</b>
18	0.676	0.688	0.675	0.529	0.656	0.682	<b>0.696</b>
20	0.677	0.698	0.594	0.560	0.720	0.699	<b>0.724</b>
22	0.647	0.656	0.625	0.557	0.679	0.662	<b>0.691</b>
24	0.631	0.622	0.564	0.592	0.656	0.638	<b>0.660</b>
26	0.603	0.613	0.541	0.562	0.630	0.622	<b>0.660</b>

Avg	0.655	0.666	0.617	0.560	0.671	0.667	<b>0.695</b>
(b)NMI							
$P$	K-means	PCA	Ncut	SC	SCC	GSC	LGSC
16	0.798	0.802	0.791	0.656	0.812	0.820	<b>0.831</b>
18	0.792	0.795	0.774	0.773	0.798	0.815	<b>0.820</b>
20	0.790	0.798	0.791	0.663	0.817	0.818	<b>0.831</b>
22	0.770	0.774	0.766	0.681	0.790	0.794	<b>0.804</b>
24	0.769	0.760	0.743	0.701	0.779	0.790	<b>0.802</b>
26	0.757	0.756	0.744	0.695	0.778	0.783	<b>0.801</b>
Avg	0.779	0.781	0.768	0.695	0.796	0.803	<b>0.815</b>

#### 4.4 USPS handwritten digit database

USPS handwritten dataset contains 9298 handwritten digit images. In this experiment, we randomly chose 200 handwritten digit images from each category as the experimental subset. Since the size of each image is  $16 \times 16$ , we resized each image to a 256-dimensional vector. Some handwritten images are shown in Fig. 2.



Fig.2 Sample images from the USPS database

Following the strategy used in the previous experiment, samples in  $P$  ( $=5, 6, \dots, 10$ ) categories were randomly selected from the subset and used to evaluate the performance of all methods. Meanwhile, we also used the same kernel function as in the previous experiments. The setting of kernel function is the same as the above experiments. We ran all methods for 10 times on each  $P$  value, and then recorded the average results in Table 3. Note that the performance of GSC is superior to that of K-means, PCA, Ncut and SC. The possible reason is that GSC takes both the manifold structure and sparsity of data into account by using the regularization technique. Compared with GSC, the proposed LGSC method further makes full use of the global geometric structure of data. Therefore, LGSC can obtain better performance than GSC in clustering.

Table 3 The clustering performance of all methods on the USPS database

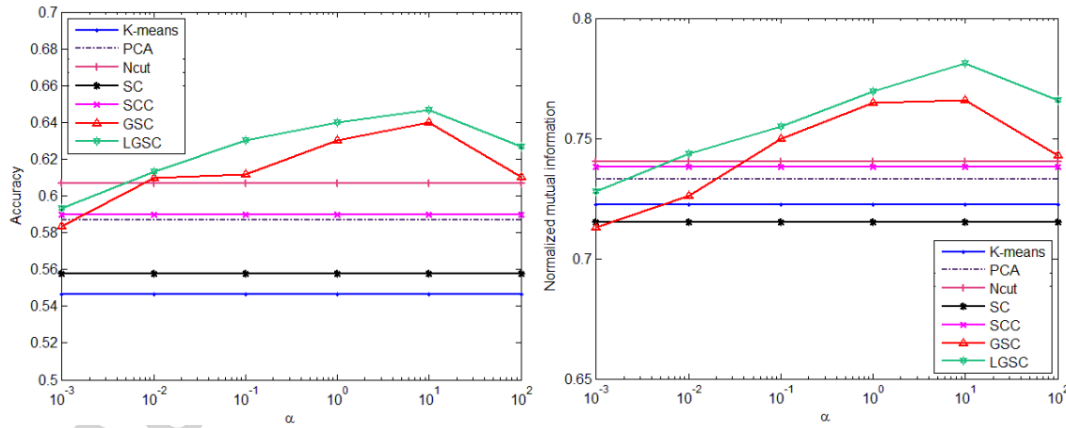
(a)AC							
$P$	K-means	PCA	Ncut	SC	SCC	GSC	LGSC
5	0.800	0.800	0.854	0.728	0.813	0.931	<b>0.954</b>
6	0.797	0.799	0.844	0.749	0.749	0.913	<b>0.925</b>
7	0.75	0.772	0.769	0.732	0.726	0.924	<b>0.939</b>
8	0.746	0.745	0.772	0.702	0.765	0.830	<b>0.849</b>
9	0.704	0.699	0.737	0.661	0.702	0.867	<b>0.885</b>
10	0.747	0.746	0.675	0.623	0.699	0.802	<b>0.811</b>
Avg	0.757	0.760	0.775	0.699	0.742	0.878	<b>0.894</b>
(a)NMI							
$P$	K-means	PCA	Ncut	SC	SCC	GSC	LGSC
5	0.659	0.660	0.824	0.654	0.673	0.835	<b>0.867</b>

6	0.639	0.641	0.816	0.688	0.686	0.842	<b>0.854</b>
7	0.634	0.643	0.793	0.634	0.642	0.843	<b>0.867</b>
8	0.630	0.629	0.784	0.621	0.667	0.807	<b>0.829</b>
9	0.608	0.608	0.776	0.551	0.641	0.810	<b>0.836</b>
10	0.629	0.628	0.748	0.545	0.649	0.763	<b>0.802</b>
Avg	0.633	0.635	0.790	0.616	0.660	0.817	<b>0.843</b>

#### 4.5 Discussion on parameter setting

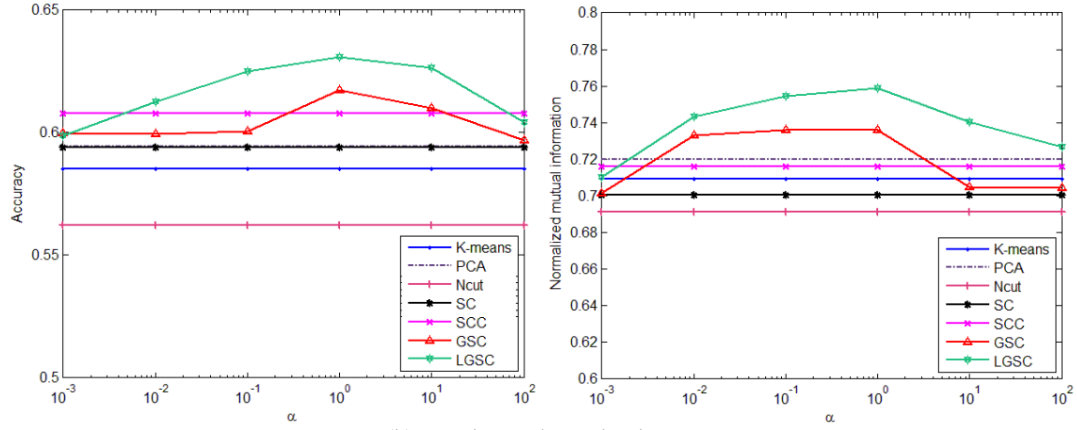
In the above experiments, the regularization parameters are empirically set to 0.1. In LGSC, there are two important parameters to be set in the objective function, i.e.  $\alpha$  and  $\mu$ . In this subsection, we present some experiments to evaluate the performance of the proposed LGSC method with various parameter values.

Fig. 3 shows the performances of all methods with different regularization parameter  $\alpha$ . In this experiment, the parameter  $\alpha$  is tuned among the set  $\{10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2\}$ . We randomly chose samples in 30, 20 and 7 categories from the ORL, the Isolet and the USPS data sets, respectively. The results in Fig.3 show that the proposed LGSC method achieves consistent performance from  $10^{-2}$  to  $10^1$ . Meanwhile, we can see that LGSC is more stable than GSC when the value of parameter  $\alpha$  varies.

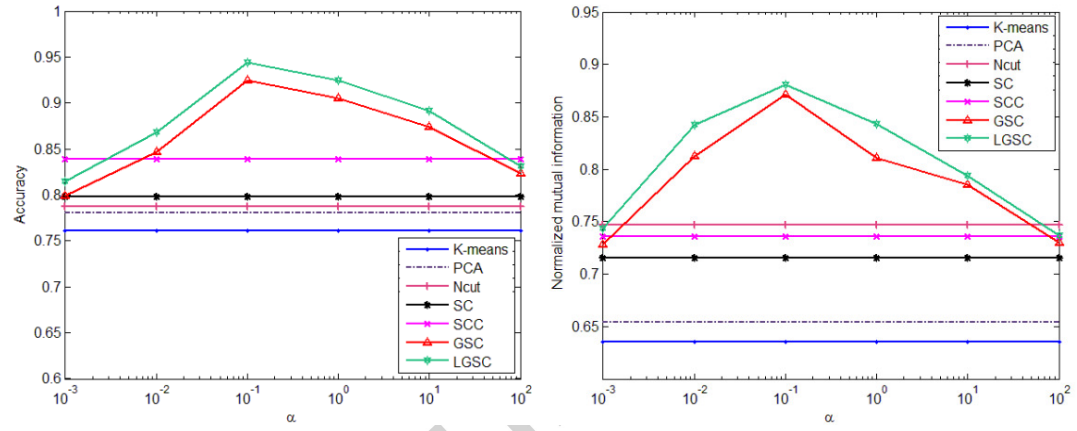


(a) Results on the ORL dataset





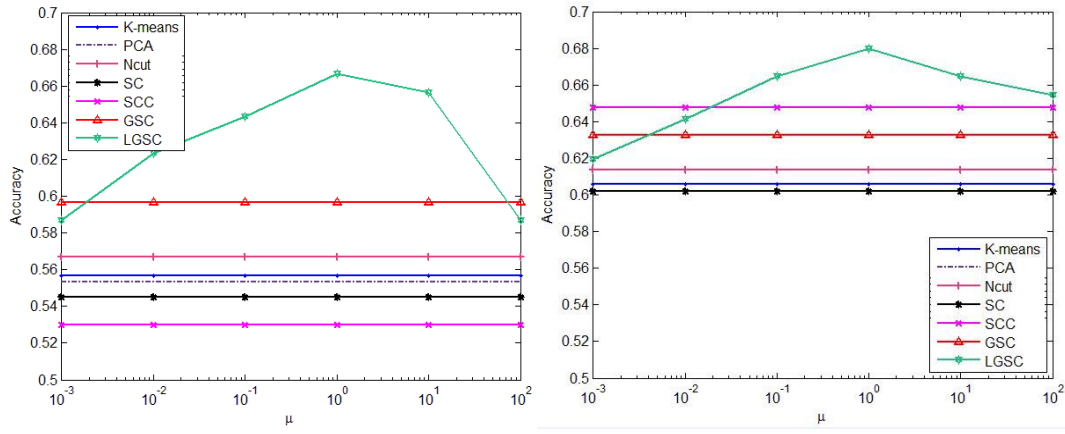
(b) Results on the Isolet dataset



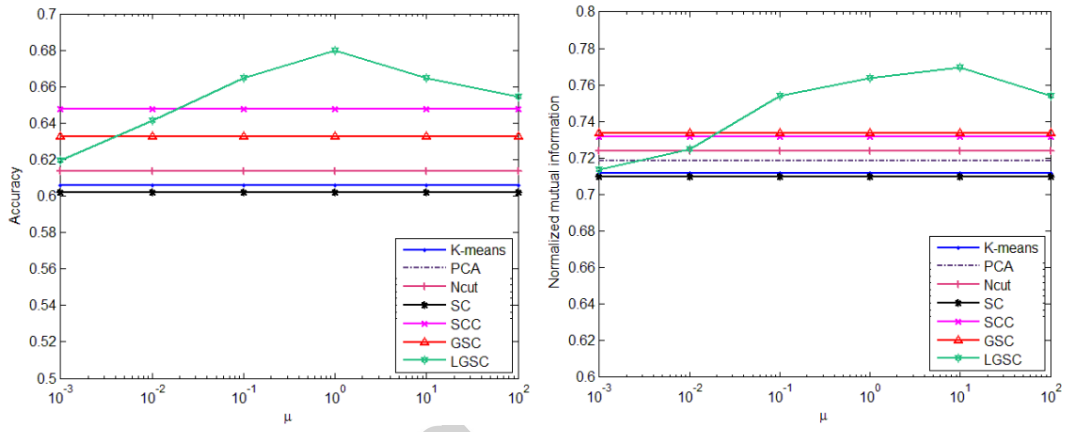
(c) Results on the USPS dataset

Fig.3 Clustering performance versus parameter  $\alpha$

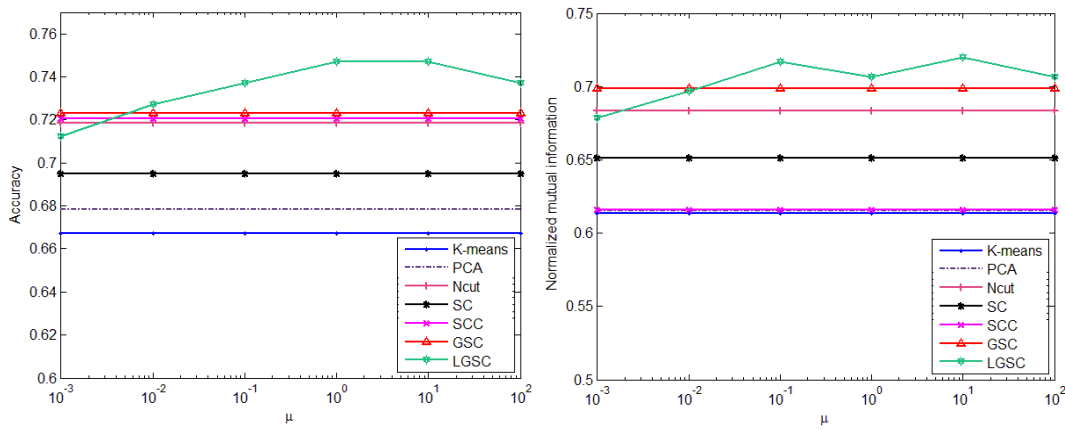
The weighting parameter  $\mu$  is used to trade off the global structure and local structure of data in the proposed model. In this subsection, we present some experiments to verify its influence to the clustering performance. The same amount of samples were randomly chosen from three datasets. Similarly, the proposed LGSC method was tuned to its best performance using different values of parameter  $\mu$ :  $10^{-3}$ ,  $10^{-2}$ ,  $10^{-1}$ ,  $10^0$ ,  $10^1$  and  $10^2$ . Fig. 4 shows the performances of all methods under different  $\mu$  values. From Fig. 4, it can be seen that the proposed method is relatively consistent under various values of the parameter  $\mu$ . The proposed LGSC method can generate the relative stable performance with a wide range of values of  $\mu$ .



(a) Results on the ORL dataset



(b) Results on the Isolet dataset

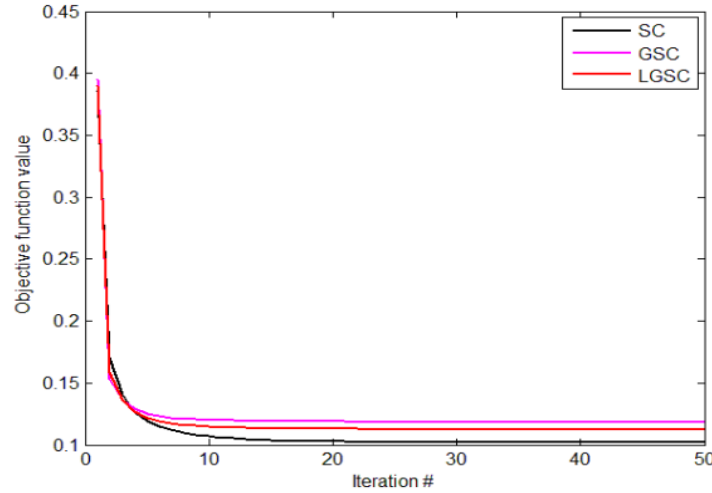


(c) Results on the USPS dataset

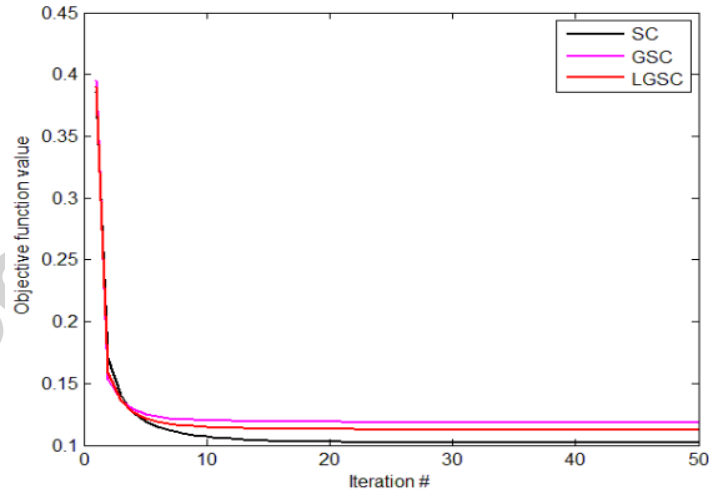
Fig.4 Clustering performance versus parameter  $\mu$

#### 4.6 The convergence of LGSC

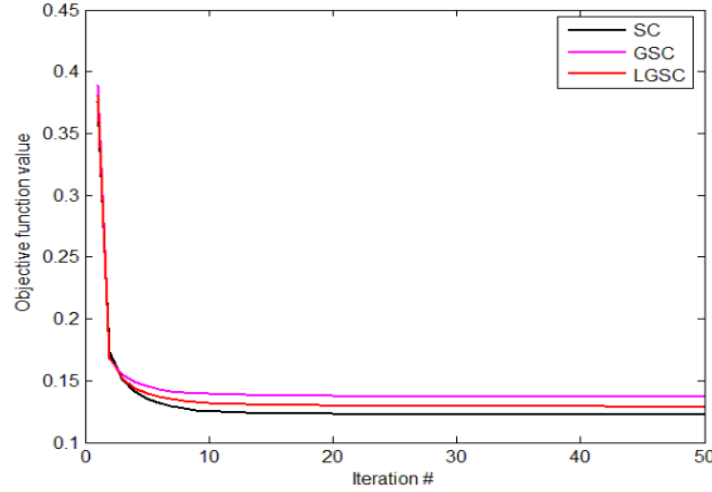
To evaluate the convergence of the proposed method, we randomly sampled from each dataset to conduct experiment. The convergence curves of SC, GSC and LGSC on three real datasets are presented in Fig. 5. In this paper, feature-sign search algorithm is used to optimize these three models. It is interesting to see from all the results that three methods can converge after 15 iterations on all cases. In addition, it can be observed that the convergence rate of the proposed LGSC is almost as fast as both SC and GSC.



(a) Performance on the ORL dataset



(b) Performance on the Isolet dataset



(c) Performance on the USPS dataset

Fig. 5 Convergence curves of SC, GSC and LGSC methods

#### 4.7 Efficiency of LGSC

We evaluated the efficiency of four sparse coding based data representation algorithms on three datasets. All experiments were conducted on a Windows 7 machine with Intel Core 2 Dual 2.10GHz CPU and 3GB RAM. In this experiment, samples from 30, 26 and 8 categories were randomly selected from the ORL, Isolet and USPS datasets for clustering.

Since SC, GSC and LGSC models are optimized by iterative update algorithms, we ran 15 iterations and listed the average time per iteration as the results. SCC can obtain the clustering result by running one time, which is reported as the final result. For all methods, we ran the experiments for 10 times and then recorded the average time cost. Table 4 shows time costs of four sparse coding based data representation methods. From Table 4, we can see that the efficiency of SCC is the best among all the methods. This is because SCC only needs to solve two regression problems, and hence is much faster than other methods which use the iterative update algorithms to optimize their models. In addition, it is clear to see that GSC and LGSC need more time to solve the optimization problem in comparison with SC. A main reason is that GSC and LGSC need to construct and compute the regularization terms. Obviously, a significant difference between GSC and LGSC is to construct a different regularization term. In GSC, we need  $O(n^3 + n^2k)$  time to construct the  $k$ -nearest neighbor graph and compute the Laplacian matrix, where  $n$  is the number of samples and  $k$  denotes the cardinality of the nearest neighbors, respectively. Meanwhile, LGSC requires  $O(n^2m + n^3k^2)$  time to construct the local and global regularization term, where  $m$  is the dimensionality of the sample. Therefore, LGSC needs a little bit more time to construct the regularization term. However, in LGSC and GSC, the time costs on constructing the regularization terms are far

less than those of the optimization procedures. Actually, it can be seen that LGSC and GSC are similar to each other in efficiency from the results in Table 4.

Table 4 The running time (in seconds) for different algorithms

Datasets	SC	SCC	GSC	LGSC
ORL	2.140	1.005	2.486	2.552
Isolet	12.150	10.598	13.667	13.824
USPS	20.013	18.412	22.488	22.521

#### 4.8 Discussion

Based on the experimental results on three real databases, we can have the following observations and discussions:

(1) As can be seen, the AC and NMI of GSC are superior to SC in all three experiments. It is reasonable because GSC incorporates the graph regularization term into traditional sparse coding method. Thus, it makes the learned sparse codes change smoothly along the geodesics of the data manifold, and the locality of the data space is preserved in low dimensional space. Compared with its competitors, GSC can provide better representation of data in new feature space.

(2) From Tables 1, 2 and 3, we can clearly see that the proposed LGSC method achieves the best performance in all cases. This is because LGSC not only considers the local manifold structure information and the sparsity of data, simultaneously, but also exploits the global structure information of data by the global regression regularizer. Thus, LGSC can capture the intrinsic geometric structure of high dimensional data. The experimental results also demonstrate the effectiveness of the proposed LGSC. Note that this observation is consistent with our motivation.

(3) For the LGSC method, two important parameters have to be set. From the experimental results on all cases, it can be seen that the proposed method can obtain relatively stable performance with a wide range of values of these parameters. Thus, they manifest that the proposed LGSC method is insensitive to these parameters.

#### 5 Conclusions and Future Work

This paper presents a novel method, called *local and global regularized sparse coding (LGSC)*, for data representation. In LGSC, the latent geometric structure of data can be discovered by the local and global regularization methods. Specifically, the local regression regularization method is used to grasp the local intrinsic structure information. Meanwhile, we employ a kernelized global regression to regularize the model and preserve the global geometric structure of data. In addition, the proposed LGSC also takes advantage of the

sparsity of data by enforcing  $l_1$ -norm regularizer on coefficient. Extensive experiments have been conducted on three benchmark datasets and have shown the effectiveness of the proposed LGSC.

However, several issues remain to be investigated in our future work. On the one hand, many norms, i.e. nuclear norm,  $l_1$ -norm, have been developed to measure the reconstruction error, which have demonstrated higher robustness than the Euclidean distance in many cases. Thus, one of the future tasks is on how to measure the error for the contaminated data. On the other hand, kernel tricks have been widely adopted in pattern recognition. A kernel version of LGSC can be easily extended from the linear version to deal with nonlinear data. Finally, another work is that the local and global regularizer will be applied for other methods, such NMF, CF and low rank representation (LRR).

#### Acknowledgments

This work was supported by the National Natural Science Foundation of China [Gran No. 61272220, 61101197], Natural Science Foundation of Jiangsu Province of China [Grant No. BK2012399, BK20140794], China Postdoctoral Science Foundation [Grant No. 2014M551599].

#### Reference

- [1]J. Yang, D. Zhang, A. F. Frangi, J.Y. Yang. Two-Dimensional PCA: a New Approach to Face Representation and Recognition, IEEE Transactions on Pattern Analysis and Machine intelligence, 2004, 26(1), 131-137.
- [2]M. M. Shafiei, S. Wang, R. Zhang. Document Representation and Dimension Reduction for Text Clustering. In Proceedings of International Conference on Data Engineering (ICDE), 2007:770-779.
- [3]Z. Li, W. Jiang, H. M. Meng. Fishervioce: A discriminant subspace framework for speaker recognition. In Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2010: 4522-4525.
- [4] P. Zhu, W. Zuo, L. Zhang et al. Unsupervised feature selection by regularized self-representation. Pattern Recognition, 2015, 48(2): 438-446.
- [5] M. Turk, A. Pentland. Eigenfaces for recognition. Journal of Neuroscience. 1991, 3(1): 71-86.
- [6] P.N. Belhumeur, J.P. Hespanha, D.J. Kriegman, Eigenfaces vs. fisherfaces: recognition using class specific linear projection, IEEE Transactions on Pattern Analysis and Machine Intelligence. 1997, 19(7): 711-720.
- [7] J.B. Tenenbaum, V. de Silva and J.C. Langford. A global geometric framework for nonlinear dimensionality reduction. Science, 2000, 290(5500): 2319-2323.
- [8] S.T. Roweis, L.K. Saul. Nonlinear dimensionality reduction by locally linear embedding. Science, 2000, 290(5500): 2323-2326.

- [9] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, *Neural Computation*. 2003, 15(6):1373-1396.
- [10] X. He, S. Yan, Y. Hu, et al. Face recognition using laplacianfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005, 27(3):328-340.
- [11] Y. Yang, F. Nie, S. Xiang, et al. Local and Global Regressive Mapping for Manifold Learning with Out-of-Sample Extrapolation. In *AAAI*, 2010.
- [12] D. Cai, X. He, J. Han, et al. Graph regularized nonnegative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2011, 33(8):1548–1560.
- [13] Q. Gu, J. Zhou, Local learning regularized nonnegative matrix factorization, In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI)*, Pasadena, California, USA, 2009.
- [14] D. Cai, X. He, J. Han, Locally consistent concept factorization for document clustering, *IEEE Transactions on Knowledge and Data Engineering*, 2011, 23(6):902–913.
- [15] Z. Shu, C. Zhao, P. Huang. Local regularization concept factorization and its semi-supervised extension for image representation. *Neurocomputing*, 2015, 158: 1-12.
- [16] J. Yang, J. Wright, T.S. Huang, et al. Image super-resolution via sparse representation. *IEEE Transactions on Image Processing*. 2010, 19(11): 2861-2873.
- [17] M. Elad, A. Michal. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 2006, 15(12): 3736-3745.
- [18] M. Julien, M. Elad, G. Sapiro. Sparse representation for color image restoration. *IEEE Transactions on Image Processing*, 2008, 17(1): 53-69.
- [19] J. Wright, J. Yang, A. Y, et al. Robust face recognition via sparse representation." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009, 31(2): 210-227.
- [20] M. Yang, L. Zhang. Gabor feature based sparse representation for face recognition with gabor occlusion dictionary. In *Proceedings of European Conference on Computer Vision (ECCV)*, Springer Berlin Heidelberg, 2010:448-461.
- [21] Z. Shu, C. Zhao, P. Huang. Constrained Sparse Concept Coding algorithm with application to image representation. *KSI Transactions on Internet and Information Systems (TIIS)*, 2014, 8(9): 3211-3230.
- [22] S. Gao, I. Tsang,; L. Chia. Laplacian Sparse Coding, Hypergraph Laplacian Sparse Coding, and Applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013, 35(1): 92-104.
- [23] D. Cai, H. Bao, X. He, Sparse concept coding for visual analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Colorado, USA, 2011:2905-2910.
- [24] S. Zhang, H. Yao, X. Sun, et al. Sparse coding based visual tracking: review and experimental comparison. *Pattern Recognition*, 2013, 46(7): 1772-1788.
- [25] Q. Li, H. Zhang, J. Guo, et al. Reference-Based Scheme Combined With K-SVD for Scene Image Categorization. *IEEE Signal Processing Letters*, 2012, 20(1): 67-70.
- [26] Q. Li Zhang, J. Guo, et al. Codebook Optimization Using Word Activation Forces for Scene Categorization. *IEEE International Conference on Image Processing (ICIP)*, 2012: 3129-3132.
- [27] H. Zou, T. Hastie, R. Tibshirani. Sparse principal component analysis. *Journal of computational and graphical statistics*, 2006, 15(2): 265-286.

- [28] Hoyer, Patrik O. Non-negative matrix factorization with sparseness constraints. *The Journal of Machine Learning Research*, 2004(5): 1457-1469.
- [29] Y. Peng, A. Ganesh, J. Wright, et al. RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012, 34(11): 2233-2246.
- [30] J. Wang, J. Yang, K. Yu, et al. Locality-constrained linear coding for image classification. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010: 3360-3367.
- [31] K. Kavukcuoglu, M. A. Ranzato, et al. Learning invariant features through topographic filter maps. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009:1605-1612.
- [32] J. Mairal, F. Bach, J. Ponce, et al. Online dictionary learning for sparse coding. In *Proceedings of 26th International Conference on Machine Learning*, 2009:689-696.
- [33] M. Zheng, J. Bu, C. Chen, et al. Graph regularized sparse coding for image representation. *IEEE Transactions on Image Processing*, 2011, 20(5):1327-1336.
- [34] I. Daubechies, M. Defriese, and C. DeMol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics*, 2004, 57:1413-1457.
- [35] S. Chen, D. Donoho, and M. Saunders. Atomic decompositions by basis pursuit. *SIAM Review*, 2001, 43:129-159.
- [36] E. Cands, J. Romberg.  $l_1$ -magic: A collection of matlab routines for solving the convex optimization programs central to compressive sampling. 2006, [www.acm.caltech.edu/l1magic/](http://www.acm.caltech.edu/l1magic/).
- [37] M. Saunders. PDCO: Primal-dual interior method for convex objectives, 2002, <http://www.stanford.edu/group/SOL/software/pdco.html>.
- [38] H. Lee, A. Battle, R. Raina, et al. Efficient sparse coding algorithms. In *Proceedings of Advance Neural Information Processing System (NIPS)*, 2007, 20:801-808.
- [39] L. Bottou, V. Vapnik. Local learning algorithms, *Neural Computation*. 1992, 4(6): 888-900.
- [40] L. Lovász and M. Plummer. *Matching Theory*. Amsterdam, the Netherlands: North Holland, 1986.



**Zhenqiu Shu** received the B.S. degree from University of South China, Hengyang, China, in 2008 and the M.S. degree Kunming University of Science and Technology, Kunming, China, in 2011. He is currently pursuing the Ph.D. degree in computer applications at Nanjing University of Science and Technology. His research interests include machine learning, data mining, and pattern recognition.

**Jun Zhou** received the B.S. degree in computer science and the B.E. degree in international business from Nanjing University of Science and Technology, Nanjing, China, in 1996 and 1998, respectively. He received the M.S. degree in computer science from Concordia University, Montreal, QC, Canada, in 2002, and the Ph.D. degree from the University of Alberta, Edmonton, Canada, in 2006.

In June 2012, he joined the School of Information and Communication Technology, Griffith University, Nathan, Qld, Australia, where he is currently a senior Lecturer. His research interests include pattern recognition, computer vision, and machine learning with human in the loop, with their applications to spectral imaging and environmental informatics.

**Pu Huang** received his B.S. and M.S. degrees in computer applications from Yangzhou University, PR China, in 2007 and 2010, respectively. He received the Ph.D. degree in Pattern Recognition and Intelligent Systems at Nanjing University of Science and Technology (NUST), China, in 2014. His research interests include pattern recognition, computer vision and machine learning.

**Zhangjing Yang** received the Ph.D. degree in Computer Application Technology at Nanjing University of Science and Technology (NUST), China, in 2014. His research interests include image process, computer vision and machine learning.

**Xun Yu** is a Ph.D. candidate at Griffith University, Brisbane, Australia. His research interests include 3D face processing and recognition, computer vision and syntactic pattern recognition.

**Chunxia Zhao** received the B.S., M.S. and Ph.D. degrees from Harbin Institute of Technology, Harbin, China, in 1985, 1988, 1998, respectively, both in the Department of Electrical Engineering and Computer. She is a professor in the Department of Computer Science, Nanjing University of Science and Technology. Her current interests are in the areas of robots, computer vision, and pattern recognition.











