Graph-oriented Learning via Automatic Group Sparsity for Data Analysis

Yuqiang Fang* Ruili Wang† Bin Dai*
*College of Mechatronic Engineering and Automation, National University of Defense Technology, Changsha, P.R. China
{fangyuqiang, daibin}@nudt.edu.cn
†School of Engineering and Advanced Technology, Massey University, Pamerston North, New Zealand
r.wang@massey.ac.nz

***Abstract*—The key task in graph-oriented learning is constructing an informative graph to model the geometrical and discriminant structure of a data manifold. Since traditional graph construction methods are sensitive to noise and less datum-adaptive to changes in density, a new graph construction method so-called ℓ^1-Graph has been proposed [1] recently. A graph construction method needs to have two important properties: sparsity and locality. However, the ℓ^1-Graph is strong in sparsity property, but weak in locality. In order to overcome such limitation, we propose a new method of constructing an informative graph using automatic group sparse regularization based on the work of ℓ^1-Graph, which is called as group sparse graph (GroupSp-Graph). The newly developed GroupSp-Graph has the same noise-insensitive property as ℓ^1-Graph, and also can successively preserve the group and local information in the graph. In other words, the proposed group sparse graph has both properties of sparsity and locality simultaneously. Furthermore, we integrate the proposed graph with several graph-oriented learning algorithms: spectral embedding, spectral clustering, subspace learning and manifold regularized non-negative matrix factorization. The empirical studies on benchmark data sets show that the proposed algorithms achieve considerable improvement over classic graph constructing methods and the ℓ^1-Graph method in various learning tasks.**

***Keywords*-graph learning; sparse representation; spectral embedding; subspace learning; non-negative matrix factorization**

I. Introduction

In recent years, manifold-based learning has become an emerging and promising approach in machine learning, with numerous recent applications in data analysis including dimensionality reduction [2], [3], [4], [5], [6], clustering [3], [7], [8] and classification [9], [1].

The main assumption in these approaches is that the data resides on a low dimensional manifold embedded in a higher dimensional space. When approximating the underlying manifold, the most common strategy is to construct an informative graph. The graph can be view as a discretized approximation of manifold sampled by the input patterns. Many manifold learning based dimensionality reduction algorithms begin by constructing an information graph. For example, ISOMAP (Isometric Feature Mapping) [2], a widely used manifold embedding method, extends metric multidimensional scaling by incorporating the geodesic distances of all pairs of measurements imposed by a global weighted graph. LE (Laplacian Eigenmaps) [3] and LLE (Locally Linear Embedding) [4] preserve proximity relationships through data manipulations on an undirected weighted graph that indicates the neighbor relations of pairwise measurements. Manifold-based clustering, e.g., spectral clustering, also can be solved by graph partitioning. Moreover, manifold subspace learning, e.g., LPP (Locality Preserving Projections) [5] and NPE (Neighborhood Preserving Embedding) [6], can be explained in a general graph framework [10]. We can see that graph plays a key role in these graph-oriented learning algorithms.

In most graph-oriented learning methods, the graph is constructed by calculating pairwise Euclidean distances, e.g., k-nearest neighbor graph. However, this graph based on pairwise distances is very sensitive to unwanted noise. To handle such problem, more recently, a new method (so called ℓ^1-Graph [1]) was proposed that constructs the graph based on a modified sparse representation framework [11], [12]. SRLP (Sparse Representation-based Linear Projections) [13] and SPP (Sparsity Preserving Projections) [8] were two new subspace learning methods which have a similar idea with ℓ^1-Graph. Both of them choose local neighborhood information for dimensionality reduction by minimizing ℓ^1 regularization objective function. Although it has shown that the ℓ^1-Graph based algorithms [1], [13], [8] outperform PCA (Principle Component Analysis) [14], LPP [5] and NPE [6] on several data sets, the ℓ^1-Graph based algorithms only have the sparsity property, but do not have the locality property (more details can be seen in Section 3.1).

In this paper, based on ℓ^1-graph, we propose a new approach to build a graph that has both sparsity and locality properties. As one known, high-dimensional data often observe sparsity and locality, which should be taken into account in the graph-oriented learning. However, in the ℓ^1-graph, the regularization term of ℓ^1 norm tends to select few bases for graph construction to favor sparsity, thus losing the locality property. Motivated by the above observations, we induce two sparse regularization methods (Elastic net [15] and OSCAR (Octagonal Shrinkage and Clustering Algorithm for Regression) [16]) that have the automatic group effect for the graph construction. Then a novel group sparse graph method (GroupSp-Graph) is proposed for several graph-oriented learning algorithms. The

1550-4786/12 $26.00 © 2012 IEEE
DOI 10.1109/ICDM.2012.82

proposed graph has the same noise-insensitive property as that of ℓ^1-graph, and also has successively preserved the group and local information in the graph. Our empirical studies on benchmark data sets demonstrate the promising results of the proposed approach.

The rest of this paper is organized as follows: the related work on graph-oriented learning algorithm is described in Section 2. In Section 3, the main disadvantage of sparse graph construction is analyzed and then the new group sparse based graph construction method is introduced for several graph-oriented learning algorithms. In Section 4, the experimental results and analysis are then presented. Finally, we conclude the discussions and indicate the further work at the end of the last section.

Notation For any vector $\boldsymbol{x}$, its transpose is denoted by $\boldsymbol{x}^\top$, its ith component by $\boldsymbol{x}[i]$. The ℓ^1-norm of $\boldsymbol{x}$ is $\|\boldsymbol{x}\|_1 = \sum_i |\boldsymbol{x}[i]|$, and its ℓ^2-norm is $\|\boldsymbol{x}\|_2 = \sqrt{\sum_i (\boldsymbol{x}[i])^2}$.

II. Related Works

A. Graph-oriented Learning

Although the motivations of different manifold and graph-oriented learning algorithms vary, their objectives are similar, which are to derive a lower-dimensional manifold representation of high-dimensional data, which can be used to facilitate the related tasks. Central to them is constructing a graph structure that models the geometrical and discriminant structure of the data manifold.

Suppose we have n data points represented as a matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n]$, where $\boldsymbol{x}_i \in \mathbb{R}^m$. With the data points, we can build a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the vertex set of the graph is referred as $\mathcal{V}(\mathcal{G}) = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n\}$, its edges set as $\mathcal{E}(\mathcal{G}) = \{e_{ij}\}$. The number of vertices of a graph $\mathcal{G}$ is its *order* [17], written as $|\mathcal{G}|$; its number of edges is denoted by $\|\mathcal{G}\|$. If an edge e_{ij} connects vertices $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, we denote the relation as $i \sim j$. The number of neighbors of a node $\boldsymbol{x}$ is called *degree* of $\boldsymbol{x}$ and is denoted by $d_\mathcal{G}(\boldsymbol{x})$, $d_\mathcal{G}(\boldsymbol{x}_i) = \sum_{i \sim j} e_{ij}$. Further, each edge e_{ij} can be weighted by $w_{ij} > 0$ for pairwise similarity measurements.

For the above notation, it is easy to see that edge e_{ij} and weight w_{ij} are important factors in graph construction. In common graph-oriented learning algorithms, the edges and weights are often specified with the following manners:

Global Graph: For $\forall i, j$, $i \sim j$, $w_{ij} = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2$ as the Euclidean distance or problem-dependent distance between two vertices, $d(\boldsymbol{x}_i) = n - 1$;

kNN Graph: For $\forall i$, $i \sim k$, $\boldsymbol{x}_k$ belongs to the k-nearest neighbor vertices for $\boldsymbol{x}_i$, $w_{ik} = \exp(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_k\|^2}{2\sigma^2})$ or $w_{ik} = 1$ simply, $d(\boldsymbol{x}_i) = k$;

ε **NN Graph**: For $\forall i$, $i \sim j$, if $\|\boldsymbol{x}_i - \boldsymbol{x}_j\| \le \varepsilon$, $w_{ij} = \exp(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\sigma^2})$ or $w_{ij} = 1$, $d(\boldsymbol{x}_i)$ is $k_\varepsilon(\boldsymbol{x}_i)$[1];

Moreover, to describe the concept of sparse graph, we induce the definition of **graph density** as follow:

Definition 1 (Graph density [18]). For an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, the graph density of $\mathcal{G}$ is $\frac{2\|\mathcal{G}\|}{|\mathcal{G}|(|\mathcal{G}|-1)}$.

From the definition, one can see the density of Global Graph is 1 and the kNN Graph has a low density $\frac{2k}{n-1}$ when $k \ll n$. Formally, we define a **dense graph** is a graph which has a high graph density, while a graph with a low graph density is a **sparse graph**.

The sparse graph plays an important role in graph-oriented learning. From the sparse graph, one can construct a matrix whose spectral decompositions reveal the low dimensional structure of the submanifold [3]. Thus, with the appropriate sparse graph, we can set up a quadratic objective function derived from the graph for embedding or subspace learning and solved by the eigenvectors of eigen-problem [10]. Also, the sparse graph based function can be incorporated as a geometric regularization in semi-supervised learning, transductive inference [9] or non-negative matrix factorization [7], [19].

B. Learning with ℓ^1-Graph

Since above traditional graph construction methods are sensitive to data noise and less datum-adaptive to changes in density, a new construct approach (so-called ℓ^1-Graph) via sparse representation has been proposed, and harnessed for prevalent graph-oriented learning tasks [1]. The motivation of ℓ^1-Graph is that each datum can be sparse reconstructed by neighbor training data. Sparse reconstructive coefficients can describe the latent local neighborhood information, which are achieved by minimizing an ℓ^1 optimization problem or lasso problem in Statistics [11].

Algorithm 1 ℓ^1-Graph Construction [1]

```
Input: Data points matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{m \times n}$
Output: Affinity matrix $\boldsymbol{W} \in \mathbb{R}^{n \times n}$
1: Normalize the data point $\boldsymbol{x}_i$ with $\|\boldsymbol{x}_i\|_2 = 1$.
2: For each data point $\boldsymbol{x}_i$, find sparse coefficient $\boldsymbol{\alpha}_i^*$ and $\boldsymbol{e}_i^*$ from the $\ell^1$ norm regularization problem:
   $$\min \|\boldsymbol{x}_i - [\boldsymbol{B}^i \ \boldsymbol{I}] \begin{bmatrix} \boldsymbol{\alpha}_i \\ \boldsymbol{e}_i \end{bmatrix} \|_2 + \lambda \| \begin{bmatrix} \boldsymbol{\alpha}_i \\ \boldsymbol{e}_i \end{bmatrix} \|_1 \quad (1)$$
   where $\boldsymbol{B}^i = [\boldsymbol{X} \setminus \boldsymbol{x}_i] \in \mathbb{R}^{m \times (n-1)}$; $\boldsymbol{I}$ is an $m$-order identity matrix, $\boldsymbol{\alpha}_i \in \mathbb{R}^{n-1}$.
3: Set affinity matrix $\boldsymbol{W}_{ij} = |\boldsymbol{\alpha}_i^*[j]|$ if $i > j$ and $\boldsymbol{W}_{ij} = |\boldsymbol{\alpha}_i^*[j-1]|$ if $i < j$.
```

In Algorithm 1, the identity matrix $\boldsymbol{I}$ is introduced as a part of dictionary to code the noise, e.g. the corrupted or occluded pixels in an image [12]. That makes ℓ^1-Graph more robust to noise than other pairwise graph construction manners. In addition, from Equation (1), we can see that

[1] $k_\varepsilon(\boldsymbol{x}_i)$ is used to denote the number of node $\boldsymbol{x}$ which satisfies $\|\boldsymbol{x}_i - \boldsymbol{x}\| \le \varepsilon$

the neighbors of $\boldsymbol{x}_i$ is automatically determined by solving an ℓ^1 regularized linear regression problem. Especially, ℓ^1 regularization as a convex relaxation of ℓ^0 regularization promotes sparsity in the solution $\boldsymbol{\alpha}_i$. Also, this sparse solution determines the set of support samples that is closest to the given sample $\boldsymbol{x}_i$. Formally, if we define the support of a sparse vector $\boldsymbol{\alpha}_i$ as $supp(\boldsymbol{\alpha}_i) = \{j : \boldsymbol{\alpha}_i[j] \neq 0\}$, the graph density of ℓ^1-Graph is $\frac{2\sum_i^n |supp(\boldsymbol{\alpha}_i)|}{n(n-1)}$. Sine $|supp(\boldsymbol{\alpha}_i)| \ll n$, ℓ^1-Graph can be defined as a sparse graph. Now, we summarize the ℓ^1-Graph construction approach as follow:

ℓ^1-**Graph**: For $\forall i$, $i \sim j$, if $\boldsymbol{\alpha}_i^*[j] \neq 0$, $w_{ij} = |\boldsymbol{\alpha}_i^*[j]|$ when $i > j$ and $|\boldsymbol{\alpha}_i^*[j-1]|$ when $i < j$, $d(\boldsymbol{x}_i) = |supp(\boldsymbol{\alpha}_i)|$.

III. Proposed Method

A. Sparsity or Group Sparsity

Research in manifold or graph-oriented learning shows that a sparse graph characterizing locality relations can convey the valuable information for classification and clustering [3]. Thus, two of important issues in graph construction are **sparsity** and **locality**.

The ℓ^1-Graph just considers sparse representation during sparse graph construction. One can choose the weights and edges connecting $\boldsymbol{x}_i$ to other vertices by solving a lasso problem, and utilize the recovery coefficients to reveal the latent locally sparsity. In our opinion, it has the following limitations:

(1) ℓ^1-norm (lasso) regularization encourages sparsity without any consideration of locality. Indeed, most graph-oriented learning algorithms are proposed under the manifold assumption. Also, the graphs in the learning algorithm are used to approximating the underlying manifold. Furthermore, the core of manifold concept is *locally Euclidean*, equivalents to the requirement that each data point $\boldsymbol{x}_i$ have a neighborhood subspace U homeomorphic to an n-ball in $\mathbb{R}^n$ [20]. Ideally, when constructing a graph via sparse coding, we desire the neighborhood subspace U is support with the data that are indicated by the nonzero sparse coefficients. That means the support samples are highly correlated with each other to satisfy the property *locally Euclidean*. So we desire the nonzero coefficients locality and sparsity not merely sparsity.

(2) ℓ^1-norm regularization encourages sparsity, but it tends to select only one sample from the entire class [21], [15], as a nearest neighbor selector in the extreme case. Thus, when some samples are correlated from different classes (e.g., digit ”7” is analogous with digit ”1” in a particular situation, but they belong to a diverse class), lasso may choose the single wrong sample to represent the test sample. So, ℓ^1-Graph is too sparse to keep the high discriminating power for graph-oriented learning.

(3) Without group constraint, the nonzero sparse coefficients by ℓ^1-norm regularization tend to unstable and the resultant model is difficult to interpret. For example, in Fig. 1, when we adjust the regularization parameter to increase the nonzero sparse coefficients, some small weight coefficients solved by lasso will be randomly distributed. However, other group sparsity regularizer, which will be mentioned in next subsection, can keep coefficients group-clustered sparse.

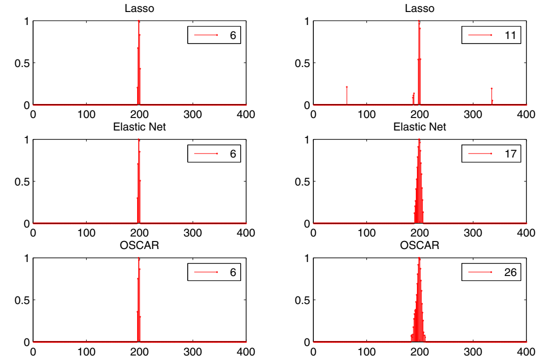


Figure 1. Nonzero sparse coefficients solved by lasso, elastic net and OSCAR; *Left*: keeping the same number of nonzero coefficients by adjusting regularization parameters; *Right*: increasing the number of nonzero coefficients to compare the performance of different regularizations. In this example, we built a dictionary with 400 noised images from the teapot data set (details in Experiment 1), then one image is reconstructed with different manners. All coefficients are normalized and shown in the above figure.

In summary, ℓ^1-norm regularization for sparse graph construction is limited, which cannot satisfy **sparsity** and **locality** constraints simultaneously. To overcome this limitation, we propose an alternate regularization method which can enforce automatic grouping sparsity.

B. Group Sparse Graph Construction

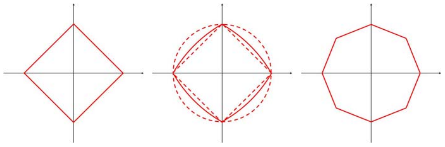
The problem of group sparsity is studied in [22], [23]. They assume that the sparse coefficients in the same group tend to be zero or nonzero simultaneously. However, in these papers, the label information of groups is required to be known in advance. In other words, they belong to supervised learning. In our method, we focus on the unsupervised learning, the same as ℓ^1-Graph. When constructing a sparse graph in an unsupervised scenario, the label information of groups can be unknown in the data set, but the sparsity and group clustering tend are known.

In this paper, two sparse regularization methods with auto-grouping effect are proposed for graph construction, elastic net [15] and OSCAR [16].

Elastic net:

Elastic net regularizer is a combination of the ℓ^1- and ℓ^2-norms. The ℓ^1 penalty promotes sparsity, while ℓ^2 penalty encourages grouping effect [15]. When applying this regularization to constructing a sparse graph, we can rewrite Equation (1) as follow:

$$\min \|\boldsymbol{x}_i - \left[\boldsymbol{B}^i\ \boldsymbol{I}\right]\boldsymbol{\beta}_i\|_2 + \lambda_1\|\boldsymbol{\beta}_i\|_1 + \lambda_2\|\boldsymbol{\beta}_i\|_2^2 \qquad (2)$$



(a) Lasso (b) Elastic Net (c) OSCAR

Figure 2. Graphical representation of constraint region of lasso, elastic net, and OSCAR; Specifically, OSCAR has an octagonal constraint region, which can achieve grouping property [16]. Elastic net can be adjusted between ℓ^1 and ℓ^2 regularization with different choices of the tuning parameter (dashed lines). Thus, elastic net could induce a grouping property with appropriate tuning.

where $\boldsymbol{\beta}_i = [\boldsymbol{\alpha}_i \ \boldsymbol{e}_i]^\top$; λ_1 and λ_2 are regularization parameters.

OSCAR:

OSCAR (Octagonal Shrinkage and Clustering Algorithm for Regression) [16] is a novel sparse model that constructs regularizer with a weighted combination of ℓ^1-norm and a pairwise ℓ^∞-norm. It encourages both sparsity and equality of coefficients for correlated samples. Thus, group sparse is automatically discovered without prior knowledge. Utilizing this regularizer, Equation (1) will be reformulated as the following optimization problem:

$$\min \|\boldsymbol{x}_i - \left[\boldsymbol{B}^i \ \boldsymbol{I}\right] \boldsymbol{\beta}_i\|_2 + \lambda_1 \|\boldsymbol{\beta}_i\|_1 + \lambda_2 \sum_{j<k} \max\{|\boldsymbol{\beta}_i[j]|, |\boldsymbol{\beta}_i[k]|\} \tag{3}$$

where λ_1 and λ_2 are regularization parameters. In Fig. 2, we can see OSCAR uses a new penalty region that is octagonal in shape, which requires no initial information regarding the grouping structure. Moreover, it can be solved efficiently based on accelerated gradient methods [24].

With the two auto-grouping regularization terms, we can discover the hidden data groups automatically and estimate the reconstruction sparse coefficient on a group-specific dictionary. Moreover, the learned data groups are consist of a small number of correlated samples. This means that locality and sparsity properties are emphasized at the same time. Formally, we can define a set G_i to indicate the nonzero regression coefficients of $\boldsymbol{x}_i$ which are solved by auto-grouping regularized sparse representation. Indeed, $G_i = supp(\boldsymbol{\alpha}_i) = \{j : \boldsymbol{\alpha}_i[j] \neq 0\}$ but further emphasizes the datum indicated by G_i belong to a neighborhood subspace and correlate with each other.

After inducing alternate regularizer for promoting group sparsity, the construction process is formally stated in Algorithm 2:

Typically, GroupSp-Graph in Algorithm 2 inherits the robustness and adaption of ℓ^1-Graph, also further emphasizes automatic group sparsity which is the lack of ℓ^1-Graph. Conveniently, we term our two GroupSp-Graph as ℓ^1/ℓ^2-graph and ℓ^1/ℓ^∞-graph individually. Both construction approaches can be summarized as follow:

Algorithm 2 GroupSp-Graph Construction

```
Input: Data points matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{m \times n}$
Output: Affinity matrix $\boldsymbol{W} \in \mathbb{R}^{n \times n}$
1: Normalize the data point $\boldsymbol{x}_i$ with $\|\boldsymbol{x}_i\|_2 = 1$.
2: For each data point $\boldsymbol{x}_i$, find sparse coefficient $\boldsymbol{\alpha}_i^*$ and $\boldsymbol{e}_i^*$ from Elastic net regularization (Equation 2) or OSCAR regularization (Equation 3)
3: Set affinity matrix $\boldsymbol{W}_{ij} = |\boldsymbol{\alpha}_i^*[j]|$ if $i > j$ and $\boldsymbol{W}_{ij} = |\boldsymbol{\alpha}_i^*[j-1]|$ if $i < j$.
```

ℓ^1/ℓ^2-Graph (**ℓ^1/ℓ^∞-Graph**): For $\forall i, i \sim j$, if $\boldsymbol{\alpha}_i^*[j] \neq 0$, $w_{ij} = |\boldsymbol{\alpha}_i^*[j]|$ when $i > j$ and $|\boldsymbol{\alpha}_i^*[j-1]|$ when $i < j$, $d(\boldsymbol{x}_i) = |G_i|$.

C. Related algorithms

In this subsection, we will integrate our GroupSp-Graph with the following graph-oriented learning algorithms for diverse tasks: data embedding, clustering, subspace learning and manifold regularized non-negative matrix factorization.

(1) Embedding via GroupSp-Graph

The Laplacian embedding algorithm [3] is a geometrically motivated spectral algorithm for efficient nonlinear dimensionality reduction or embedding. Laplacian embedding can preserve the local topology of original data in the embedded space through an informative graph. Also, the embedded result can be solved with the eigen-problem of a graph laplacian matrix. Typically, when using the group sparse graph for spectral embedding, the affinity matrix is automatically constituted with reconstruction coefficients α_i^*. Moreover, the group sparse coefficients are significative to emphasize "local topology". The detailed algorithm based on GroupSp-Graph is listed in Algorithm 3.

Algorithm 3 Spectral Embedding via GroupSp-Graph

```
Input: Data points matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{m \times n}$
Output: Embedding data points matrix $\boldsymbol{Y} \in \mathbb{R}^{d \times n}$, $d \ll m$
1: Constructing the Graph via Algorithm 2
2: Symmetrize affinity matrix $\boldsymbol{W} = (\boldsymbol{W} + \boldsymbol{W}^\top)/2$
3: Set Laplacian matrix $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$, where $\boldsymbol{D}_{ii} = \sum_j \boldsymbol{W}_{ij}$
4: Compute eigenvalues and eigenvectors for the generalized eigenvector problem:$\boldsymbol{L}\boldsymbol{y} = \lambda \boldsymbol{D}\boldsymbol{y}$
5: Let $\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_n$ be the eigenvectors, sorted in increasing according to each eigenvalues. Then the embedding data points matrix is given by $\boldsymbol{Y} = [\boldsymbol{y}_2, \cdots, \boldsymbol{y}_{d+1}]^\top$
```

(2) Subspace Learning via GroupSp-Graph

SPP (Sparsity Preserving Projections) [8] is a recent subspace learning algorithm, in which the learning process is essentially achieved by constructing ℓ^1-Graph. Furthermore in [1], ℓ^1-Graph is also used for a subspace learning. Based on the same notion, in this subsection, we develop a subspace learning algorithm with GroupSp-Graph.

Basically, the generic problem subspace learning is to find a transformation matrix $\boldsymbol{A} \in \mathbb{R}^{m \times d}$ that maps each point $\boldsymbol{x}_i \in \mathbb{R}^m$ to a low dimension represent $\boldsymbol{y}_i \in \mathbb{R}^d (d \ll m)$, where $\boldsymbol{y}_i = \boldsymbol{A}^\top \boldsymbol{x}_i$. The purpose of transformation matrix $\boldsymbol{A}$ can be formulated as the following objective problem:

$$\min \sum_i \|\boldsymbol{A}^\top \boldsymbol{x}_i - \sum_j \boldsymbol{W}_{ij} \boldsymbol{A}^\top \boldsymbol{x}_j\|_2^2 \tag{4}$$

This problem can be solved by the generalized eigenvector problem [5], [6]. Now, if we use GroupSp-Graph to construct affinity $\boldsymbol{W}$, the new subspace learning can be instead summarize as in Algorithm 4:

```
Algorithm 4 Subspace Learning via GroupSp-Graph
Input: Data points matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{m \times n}$
Output: Transformation matrix $\boldsymbol{A} \in \mathbb{R}^{m \times d}$
1: Constructing the Graph via Algorithm 2
2: Solve the generalized eigenvector problem:
       $\boldsymbol{X}\boldsymbol{M}\boldsymbol{X}^\top \boldsymbol{a} = \lambda \boldsymbol{X}\boldsymbol{X}^\top \boldsymbol{a}$        (5)
   where $\boldsymbol{M} = (\boldsymbol{I} - \boldsymbol{W})^\top (\boldsymbol{I} - \boldsymbol{W}); \boldsymbol{I} = diag(1, \cdots, 1)$
3: Let $\boldsymbol{a}_1, , \boldsymbol{a}_1, \cdots, \boldsymbol{a}_d$ be the eigenvectors, sorted according to each eigenvalues $\lambda_1 \leq, \cdots, \lambda_d$. Then the transformation $\boldsymbol{A}$ is given by $\boldsymbol{A} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_d] \in \mathbb{R}^{m \times d}$
```

(3) Non-negative Matrix Factorization via GroupSp-Graph

The Non-negative Matrix Factorization(NMF) is a popular algorithm to learn the parts of the data representation, e.g., faces and text documents [25]. Recently, graph or manifold regularization is incorporated into the non-negative matrix factorization, named Graph-regularized Non-negative Matrix Factorization (GNMF) [7]. The GNMF received the state-of-the-art performance due to build a new parts-based representation space which respects the geometrical structure of the data space. In this subsection, following the idea of GNMF, we utilize our GroupSp-Graph as a regularization to non-negative matrix factorization.

Considering the data points matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{m \times n}$, NMF aims to find two non-negative matrices $\boldsymbol{U} = [\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots, \boldsymbol{u}_r] \in \mathbb{R}^{m \times r}$, and $\boldsymbol{V} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_n] \in \mathbb{R}^{r \times n}$ such that $\boldsymbol{X} \approx \boldsymbol{U}\boldsymbol{V}$. Usually hidden factor r is chosen to be smaller than n or m. Thus, a compressed approximation can be rewritten column by column as $\boldsymbol{x}_i = \boldsymbol{U}\boldsymbol{v}_i, i = 1, 2, \cdots, n$. Therefore, $\boldsymbol{U}$ can be regarded as containing a basis that is optimized for linear combination of the data in $\boldsymbol{X}$. In GNMF [7], [19], by integrating the manifold regularization [9], GNMF minimizes the objective function as follows:

$$\min \|\boldsymbol{X} - \boldsymbol{U}\boldsymbol{V}\|_F^2 + \lambda \mathrm{Tr}(\boldsymbol{V}\boldsymbol{L}\boldsymbol{V}^\top) \tag{6}$$

where $\|\cdot\|_F$ denotes the Frobenius norm; $\mathrm{Tr}(\cdot)$ is the trace of matrix and $\boldsymbol{L}$ is the graph laplacian matrix. The detailed algorithm based on GroupSp-Graph is described in Algorithm 5.

```
Algorithm 5 Non-negative Matrix Factorization via GroupSp-Graph
Input: Data points matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{m \times n}$, hidden factor $r$, and $r \ll n$, $r \ll m$;
Output: Non-negative basis matrix $\boldsymbol{U} \in \mathbb{R}^{m \times r}$ and coefficient matrix $\boldsymbol{V} \in \mathbb{R}^{r \times n}$;
1: Constructing the Graph via Algorithm 2
2: Symmetrize affinity matrix $\boldsymbol{W} = (\boldsymbol{W} + \boldsymbol{W}^\top)/2$
3: Set Laplacian matrix $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$, where $\boldsymbol{D}_{ii} = \sum_j \boldsymbol{W}_{ij}$
4: Solve the regularized optimization problem:
       $\min_{\boldsymbol{U}, \boldsymbol{V}} \|\boldsymbol{X} - \boldsymbol{U}\boldsymbol{V}\|_F^2 + \lambda \mathrm{Tr}(\boldsymbol{V}\boldsymbol{L}\boldsymbol{V}^\top)$        (7)
   where $\lambda$ is the regularized parameter.
5: Approximation of data $\boldsymbol{x}_i = \boldsymbol{U}\boldsymbol{v}_i, i = 1, 2, \cdots, n$
```

IV. Experiments

In this section, we evaluate GroupSp-Graph with different learning tasks, including data embedding, clustering, subspace learning and non-negative matrix factorization. Furthermore, we solve the sparse modeling problems: lasso, elastic net and OSCAR in the same framework with accelerated gradient methods [26], [24].

A. Spectral Embedding

In this experiment, we compare our GroupSp-Graph based spectral embedding algorithm with Laplacian Eigenmaps and ℓ^1-Graph based spectral embedding algorithm. In the experiment, the teapot database is used, which contains 400 teapot color images (each of size 76×101×3). The teapot was viewed in full 360 degrees of rotation. Theoretically, two-dimensional embedding of the database should be a circular, which reflects the underlying rotational degree of freedom [27]. In addition, noise has been added to each image to demonstrate the proposed algorithm is insensitive to data noise. The results are shown in Fig. 3.

As one can see, embedding with kNN-Graph does not succeed in unraveling the manifold and recovering the two underlying degrees of freedom due to the influence of noise as shown in Fig. 3(a). Although a reliable embedding can be obtained by ℓ^1-Graph embedding with a small parameter λ, the results are not stable when adjusting sparse parameter to increase nonzero coefficient as shown in Fig. 3(b). In contract, a reliable and stable embedding manifold is obtained by our proposed methods (ℓ^1/ℓ^2-Graph and ℓ^1/ℓ^∞-Graph) with different number of nonzero sparse coefficients. Also, the two-dimensional embedding approximates a circular as shown in Fig. 3(c) and (d).

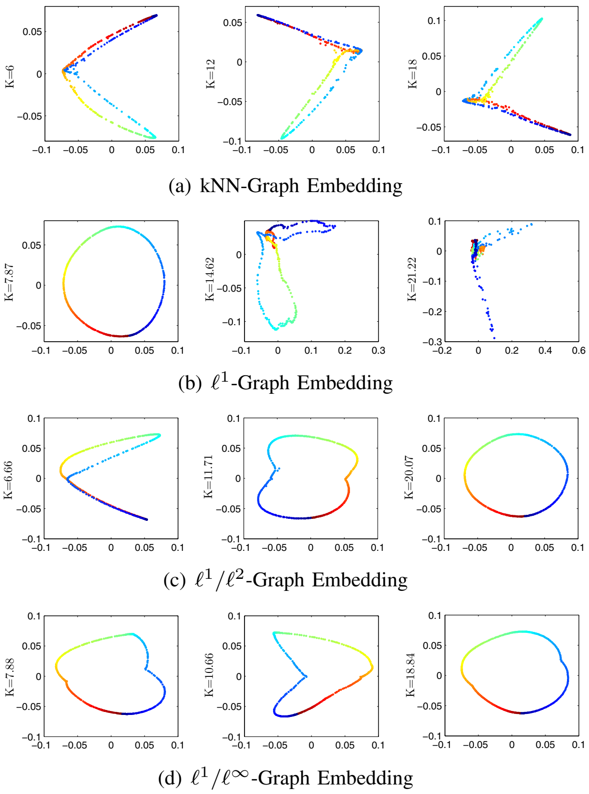


Figure 3. 2D embedding of the teapot dataset obtained by: (a) kNN-Graph (b) ℓ^1-Graph (c) ℓ^1/ℓ^2-Graph (d) ℓ^1/ℓ^∞-Graph. For visualization purpose, color coding is used to reveal how the data is embedding in two dimensions. The results are also computed for different choices of the number of nearest neighbors K. K in (b), (c), (d) are the average of the number of nonzero sparse coefficients

B. *Spectral Clustering*

In this subsection, we investigate the performance of our proposed approach in spectral clustering. Spectral clustering is an unsupervised learning task and we compare our method with PCA (Principal Component Analysis) [14] and several different graph-based methods, e.g., Laplacian Eigenmaps [3] and ℓ^1-Graph based cluster method [1]. We choose K-means as our basic clustering algorithm. K-means is performed in the reduced feature space by PCA and other graph-oriented embedding methods. For visualization, the reduced dimension is set to be three. In this experiments, 200 randomly selected samples of each digit (i.e., 1, 2 and 3) from USPS handwritten digit database [28] are used and the images are normalized to the size of 32× 32 pixels. Following the approach in [1], we use two standard metrics, the accuracy (ACC) and the normalized mutual information (NMI), to measure the clustering performance. Both ACC and NMI range from 0 to 1, while ACC reveals the clustering accuracy and NMI indicates whether the different clustering sets are identical (NMI=1) or independent (NMI=0). The detail about the these metrics can be found in [1] and [29].

Table I
PERFORMANCE COMPARISONS ON THE EXTENDED YALE B DATABASE.(NOTE: OUR(1) DENOTES ℓ^1/ℓ^2-GRAPH), OUR(2) DENOTES ℓ^1/ℓ^∞-GRAPH).

Accuracy						
d^*	PCA[14]	NPE[6]	LPP[5]	SPP[8]	Our(1)	Our(2)
50	0.7882	0.8968	0.8392	0.8898	**0.9311**	0.9171
100	0.7351	0.8104	0.7342	0.8985	**0.9147**	0.9114
200	0.5617	0.8370	0.6426	0.7746	**0.9054**	0.8880
500	0.8147	0.8264	0.8614	0.8499	0.8829	**0.8851**
Avg.	0.7249	0.8427	0.7693	0.8532	0.9085	**0.9029**

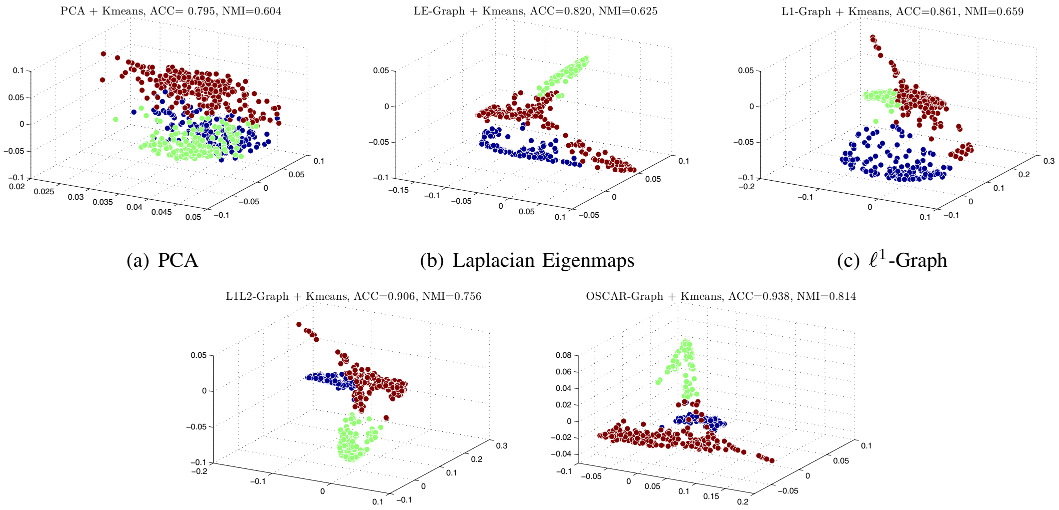
$^*d$ is the reduced dimensionality

Fig. 4 shows the visualization of the clustering results. The images of digits (i.e., 1, 2 and 3) from the USPS database are mapped into a 3-dimensional space and then clustered with K-means. As shown in the figure, compared with PCA, LE (Laplacian Eigenmaps) and ℓ^1-Graph got good results by preserving the embedded geometry structure. However, the better results are obtained by ℓ^1/ℓ^2-Graph and ℓ^1/ℓ^∞-Graph, where the data are much better separated by taking clustered sparsity into consideration in graph. Meantime, the proposed GroupSp-Graph based spectral clustering algorithm is better than the other evaluated algorithms for two qualitative metrics: ACC and NMI.

C. *Subspace Learning*

In this experiment, we compare GroupSp-Graph based subspace learning algorithm with several widely used unsupervised subspace learning techniques for face recognition [14], [5], [6], [8]. The Extended Yale B [30] face database is used in this test. It consists of a total of 38 individuals (64 samples per person). Each image is normalized to the size of 32×32 pixels. To evaluate the algorithmic performance on the database, we randomly select 50 images for each individual and the rest are used for testing. Here, the recognition rates are chosen to measure the performance. For the baseline method, we simply performed face recognition in the original 1024-dimensional image space, and the baseline recognition rate is 0.8415. We use the classical nearest neighbor classifier for comparing the discriminating power from each subspace learning approach. The best results obtained in the different subspaces and the corresponding dimensionality (50, 100, 200, 500) for each method are shown in Table 1 and Fig. 5.

In the test, we choose an optimal parameter over a range (parameters: lasso, λ_1=0.001; elastic net: λ_1=0.0001, λ_2=0.0001; OSCAR: λ_1=0.0001, λ_2=0.0001). In general, the performance of different methods varies with the number of dimensions. From the recognition rates summarized in Table 1, one can see the best results obtained by proposed GroupSp-Graph based subspace learning algorithms.



(a) PCA (b) Laplacian Eigenmaps (c) ℓ^1-Graph

(d) ℓ^1/ℓ^2-Graph (e) ℓ^1/ℓ^∞-Graph

Figure 4. Visualization of the clustering results. (a) PCA (b) Laplacian Eigenmaps (c) ℓ^1-Graph (d) ℓ^1/ℓ^2-Graph and (e) ℓ^1/ℓ^∞-Graph algorithm for three cluster (handwritten digits 1, 2, and 3 in USPS database). Various colors of the points indicate different digits. Two compared metrics (ACC and NMI) are listed above the figures.

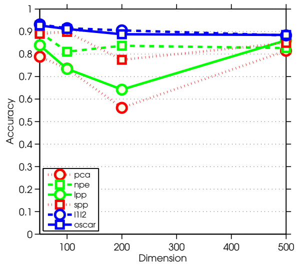


Figure 5. Performance comparisons on the Extended Yale B database.

D. Non-negative Matrix Factorization

NMF (Non-negative Matrix Factorization) is known as a powerful tool for data reduction and clustering which achieves the state-of-the-art performance. In this subsection, we evaluate proposed NMF via GroupSp-Graph (GS-GNMF) algorithm on image clustering task. To demonstrate that the clustering performance can be improved by our method, our method is compared with Canonical K-means, NMF [25] and GNMF (Graph regularized Non-negative Matrix Factorization) [7], [19]. The COIL20 data set [31] is used in our experiment, which contains 32 × 32 gray scale images of 20 objects viewed from varying angles. Each object has 72 images from different viewpoints. To measure the clustering performance, we also use two standard metrics, the accuracy (ACC) and the normalized mutual information (NMI).

For the baseline method, we simply performed K-means in the original image space. In the test, GNMF has two parameters: the number of nearest neighbors k and the regularization parameter λ. Following the suggestion in [19], we set k to 5 and λ to 100. In GS-GNMF, we use elastic net as a group sparse regularization and empirically set the parameters λ_1=0.0001 and λ_2=0.001. Table 2 and Fig. 6 show the clustering results on COIL20 with different hidden factor r. One can see that both GS-GNMF and GNMF result the best performance by inducing the graph structure. Especially, GS-GNMF achieves slightly better performance than GNMF.

When we add noise to the images, the performance of GNMF decreased drastically as σ increases (σ is the factor in the noise model[2] to control the noise level), as shown in Fig. 7. This is because GNMF uses k-nearest neighbor graph to capture the local geometric structure, which is sensitive to noise. However, the performance of GS-GNMF decreases slightly when σ increases as shown in Fig. 7. This demonstrates GS-GNMF is not so sensitive to noise.

V. Conclusion

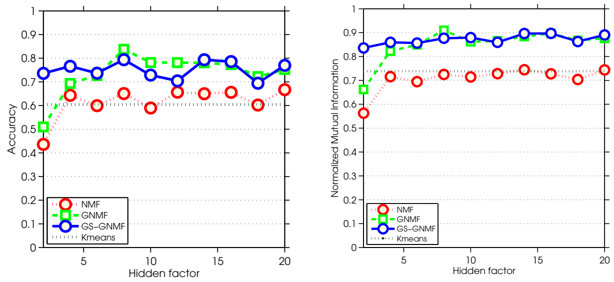
In this paper, we have explored the novel method of constructing an information graph using automatic group sparse regularization, which is called as group sparse graph (GroupSp-Graph). The GroupSp-Graph is an extension of ℓ^1-Graph by integrating the properties of sparsity and locality simultaneously. Also, we integrate the group sparse graph with various graph-oriented learning algorithms: spectral embedding, spectral clustering, subspace learning and non-negative matrix factorization. The experimental results on

[2] In the experiment, we choose salt and pepper noise to corrupt the images in COIL20 data set.

Table II
Clustering performance comparisons on COIL20.

	ACC			NMI		
r^*	NMF	GNMF	Our	NMF	GNMF	Our
2	0.4361	0.5097	**0.7361**	0.5629	0.6622	**0.8360**
4	0.6431	0.6931	**0.7659**	0.7153	0.8226	**0.8594**
6	0.5991	0.7270	**0.7368**	0.6947	0.8491	**0.8562**
8	0.6506	**0.8375**	0.7938	0.7247	**0.9097**	0.8763
10	0.5896	**0.7819**	0.7285	0.7143	0.8617	**0.8793**
12	0.6562	**0.7815**	0.7044	0.7281	**0.8658**	0.8593
14	0.6500	0.7805	**0.7936**	0.7445	0.8837	**0.8962**
16	0.6556	0.7729	**0.7854**	0.7274	**0.8966**	0.8965
18	0.6020	**0.7222**	0.6944	0.7037	**0.8674**	0.8630
20	0.6673	0.7522	**0.7701**	0.7436	0.8759	**0.8902**
Avg.	0.6105	0.7359	**0.7509**	0.7059	0.8495	**0.8712**

[*] r is the hidden factor

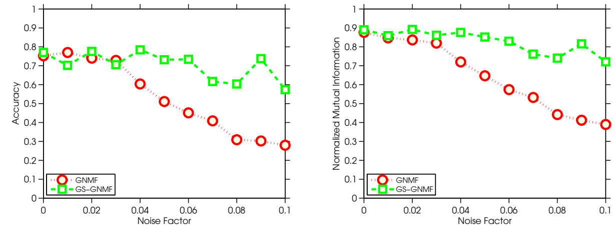


(a) ACC

(b) NMI

Figure 6. (a) Accuracy (b) Normalized Mutual Information. Clustering performance comparisons on COIL20

each task and data sets show that the proposed algorithm achieves considerable improvements over traditional graph construction methods [14], [6], [5], [3] and the ℓ^1-Graph method [8], [1]. Furthermore, since graph is widely used in computer vision and machine learning, our technique can be applied in other latest graph-oriented learning algorithms, e.g., graph regularized sparse coding [32] and graph-based ranking for image retrieval [33].



(a) ACC

(b) NMI

Figure 7. (a) Accuracy (b) Normalized Mutual Information. Clustering performance comparisons between GNMF and GS-GNMF on COIL20 under different noise conditions

VI. Acknowledgement

The authors would like to thank the anonymous referees for their helpful comments and suggestions. This research is supported by the National Nature Science Foundation of China (61075043).

References

[1] B. Cheng, J. Yang, S. Yan, Y. Fu, and T. Huang, "Learning with ℓ^1-graph for image analysis," *IEEE Transcations on Image Processing*, vol. 19, no. 4, pp. 858–866, 2010.

[2] J. Tenenbaum, V. Silva, and J. Langford, "A glboal geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[3] M. Belkin and M. Niyogi, "Laplacian eigenmaps for dimemsionality reduction and data representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[4] R. Sam and S. Lawrence, "Nonlinear dimenionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[5] X. He and P. Niyogi, "Locality preserving projections," in *Proceedings of Advances in Neural Information Processing Systems 16*, pp. 9–16, MIT Press, 2003.

[6] X. He, D. Cai, S. Yan, and H. Zhang, "Neighborhood preserving embedding," in *Proceedings of the International Conference on Computer Vision*, pp. 17–21, IEEE, 2005.

[7] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized non-negative matrix factorization for data representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1548–1560, 2011.

[8] L. Qiao, S. Chen, and X. Tan, "Sparsity preserving projections with applications for face recognition," *Pattern Recognition*, vol. 43, no. 1, pp. 331–341, 2010.

[9] B. Milhail, N. Partha, and S. Vikas, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *Journal of Machine Learning Research*, vol. 7, no. 48, pp. 2399–2434, 2006.

[10] S. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 40–51, 2007.

[11] T. Robert, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society, Series B*, vol. 58, no. 1, pp. 267–288, 1996.

[12] J. Wright, A. Yang, A. Ganesh, S. Sastry, and M. Yi, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[13] R. Timofte and L. Van Gool, "Sparse representation based projections," in *Proceedings of the 22th British Machine Vision Conference*, pp. 61.1–61.12, BMVA Press, 2011.

[14] H. Hotelling, “Analysis of a complex of statistical variables into principal components,” *Journal of Educational Psychology*, vol. 29, no. 1, pp. 40–51, 1993.

[15] H. Zou and H. Hastie, “Regression and variable selection via the elastic net,” *Journal of the Royal Statistical Society, Series B*, vol. 67, no. 2, pp. 301–320, 2005.

[16] H. Bondell and B. Reich, “Simultaneous regression shrinkage, variable selection and supervised clustering of predictors with oscar,” *Biometrics*, vol. 64, no. 1, pp. 115–123, 2008.

[17] D. Reinhard, *Graph Theory*. Heidelberg, Germany: Springer-Verlag, 2010.

[18] F. Coleman, Thmas and J. More, Jorge, “Estimation of sparse jacobian matrices and graph coloring problems,” *SIAM Journal on Numerical Analysis*, vol. 20, no. 1, pp. 187–209, 1983.

[19] D. Cai, H. Xiaofei, X. Wu, and J. Han, “Non-negative matrix factorization on manifold,” in *Proceedings of the International Conference on Data Mining*, pp. 63–72, IEEE, 2008.

[20] S. Lang, *Introduction to Differentiable Manifolds*. Heidelberg, Germany: Springer-Verlag, 2002.

[21] A. Majumdar and R. Ward, “Classification via group sparsity promoting regularization,” in *Proceedings of the 32th International Conference on Acoustics,Speech and Signal processing*, pp. 861–864, IEEE, 2009.

[22] M. Yuan and Y. Lin, “Model selection and estimation in regression with grouped variables,” *Journal of the Royal Statistical Society, Series B*, vol. 68, no. 1, pp. 49–67, 2006.

[23] M. Stojnic, F. Parvaresh, and B. Hassibi, “On the reconstruction of block-sparse signals with an optimal number of measurements,” *IEEE Transaction on Signal Processing*, vol. 57, no. 8, pp. 3075–3085, 2009.

[24] J. Kwok and W. Zhong, “Efficient sparse modeling with automatic feature grouping,” in *Proceedings of the 28th International Conference on Machine Learning*, pp. 9–16, IEEE, 2011.

[25] L. Daniel D and H. S. Seung, “Learning the parts of objects by non-negative matrix facrization,” *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[26] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm for linear inverse problems,” *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[27] Weinberger and Saul, “An introduction to nonlinear dimensionality reduction by maximum variance unfolding,” in *Proceedings of National Conference on Artificial Intelligence*, pp. 1683–1686, ACM, 2006.

[28] J. Hull, “A database for handwritten text recognition research,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 5, pp. 550–554, 1994.

[29] X. Zheng, D. Cai, X. He, W. Ma, and X. Lin, “Locality preserving clustering for image database,” in *Proceedings of ACM International Conference on Multimedia*, (New York, USA), pp. 885–891, ACM, 2004.

[30] A. Georghiades, P. Belhumeur, and D. Kriegman, “From few to many: Illumination cone models for face recognition under variable lighting and pose,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 643–660, 2001.

[31] S. Nene, S. Nayar, and H. Murase, “Columbia objec image library(coil-20),” *Technical Report CUCS-005-96 Columbia University*, 1996.

[32] M. Zheng, J. Bu, C. Chen, C. Wang, L. Zhang, G. Qiu, and D. Cai, “Graph regularized sparse coding for image representation,” *IEEE Transactions on Image Processing*, vol. 20, no. 5, pp. 1327–1336, 2011.

[33] X. Bin, B. Jiajun, C. Chun, C. Deng, H. Xiaofei, L. Wei, and L. Jiebo, “Efficient manifold ranking for image retrieval,” in *Proceedings of the 34th Annual International ACM SIGIR Conference*, pp. 885–891, ACM, 2011.