

Research Article

A Rank-Constrained Matrix Representation for Hypergraph-Based Subspace Clustering

Yubao Sun,¹ Zhi Li,¹ and Min Wu²

¹*Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment Technology, B-DAT Lab, School of Information & Control, Nanjing University of Information Science and Technology, Nanjing 210014, China*

²*Department of Medical Engineering, Nanjing General Hospital of Nanjing Area Command, Nanjing University School of Medicine, Jinling Hospital, Nanjing 210002, China*

Correspondence should be addressed to Min Wu; njzywm@163.com

Received 5 July 2015; Accepted 8 September 2015

Academic Editor: Giovanni Garcea

Copyright © 2015 Yubao Sun et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a novel, rank-constrained matrix representation combined with hypergraph spectral analysis to enable the recovery of the original subspace structures of corrupted data. Real-world data are frequently corrupted with both sparse error and noise. Our matrix decomposition model separates the low-rank, sparse error, and noise components from the data in order to enhance robustness to the corruption. In order to obtain the desired rank representation of the data within a dictionary, our model directly utilizes rank constraints by restricting the upper bound of the rank range. An alternative projection algorithm is proposed to estimate the low-rank representation and separate the sparse error from the data matrix. To further capture the complex relationship between data distributed in multiple subspaces, we use hypergraph to represent the data by encapsulating multiple related samples into one hyperedge. The final clustering result is obtained by spectral decomposition of the hypergraph Laplacian matrix. Validation experiments on the Extended Yale Face Database B, AR, and Hopkins 155 datasets show that the proposed method is a promising tool for subspace clustering.

1. Introduction

High-dimensional data spaces are frequently encountered in computer vision and machine learning tasks. In most cases, the data points lie in multiple low-dimensional subspaces embedding in a high-dimensional ambient space, and their intrinsic dimension is often much smaller than the dimension of the ambient space [1, 2]. When the subspace structure and membership of the data points to the subspaces are unknown, it is necessary to cluster the data into multiple subspaces. Subspace clustering is therefore of use in computer vision (e.g., image segmentation, motion segmentation, and face clustering), machine learning, and image analysis [3, 4].

Over the past twenty years, several subspace clustering methods [5] have been proposed. The existing methods can be roughly divided into four categories: factorization methods [6–8], algebraic methods [9, 10], statistical methods [11–13], and sparse methods [14–17]. In matrix factorization-based algorithms, a similarity matrix is built by factorization

of the data matrix, followed by spectral clustering of the similarity matrix. This method assumes that the subspaces are independent and the data are clean [6–8]. Thus, these methods cannot cope well with nonindependent subspaces structure and their performance degenerates with noisy data. Algebraic methods utilize the structure of subspaces for clustering. Generalized principal component analysis (PCA) [9, 10] is the archetypal algebraic method. It does not assume that linear subspaces are independent or disjoint, but its complexity increases exponentially with the number of samples and the dimensions of the subspaces. In statistical methods, a distribution model is defined for the data drawn from the subspaces prior to estimating the model parameters using statistical inference. Mixture of Probabilistic PCA (MPPCA) [11] utilizes a mixture of Gaussians to represent the a priori probability of the data. The Agglomerative Lossy Compression (ALC) algorithm [12] assumes that the data are drawn from a mixture of degenerate Gaussians. RANSAC [13] uses a greedy strategy for labeling data as inliers and

outliers, before iteratively fitting the sampled point into the statistical model and updating the parameters based on the residual. Sparse-based methods utilize low-rank and sparse properties of the data for subspace clustering. Sparse Subspace Clustering (SSC) [14] represents a data point as the sparse combination of all other data points in the set by minimizing the l_1 norm of the coefficients. The low-rank representation (LRR) algorithm [15, 16] aims to find a low-rank representation of the data matrix, with the minimization of nuclear norm to constrain the sum of the singular values of the coefficient matrix. Different from sparse representation, LRR represents a data sample as a linear combination of the atoms in a dictionary and jointly constrains the low-rank property of all the coefficients of the sample set, so it captures the global structure of the data [15, 17]. Due to this advantage, LRR recently attracts much attention.

The LRR algorithm uses the relaxed convex model to find the low-rank representation of the data, with exact decomposition of the data matrix into low-rank and sparse error components. However, when data are noisy, an exact low-rank and sparse matrix decomposition does not always exist for an arbitrary matrix [18, 19]. Furthermore, the rank range of the LRR model cannot be directly controlled and, in some cases, the range of the rank of the representation needs to be explicitly constrained. For example, in a face recognition problem, the images of the face of an individual under different lighting conditions can be simply characterized as a nine-dimensional linear subspace in the space of all possible images [20]. In a motion segmentation problem, if n objects move independently and arbitrarily in a 3D space, then the motion trajectories lie in n independent affine subspaces of three dimensions [5]. Thus, these prior ranges can be used as the upper bound to construct an efficient and rank-constrained representation, when the face images or motion trajectories are corrupted by lighting variations or outliers.

In the LRR model, the low-rank representation is used to define the undirected weighted pairwise graph for spectral clustering. In fact, the large coefficients in the low-rank representation usually cluster in groups following the analysis results of [15, 16]. It is implied that group information among the data is useful for clustering besides the pairwise relationship between two samples. However, the conventional pairwise graph, as in [15, 16], fails to effectively describe the complex correlations that exist between samples [21, 22].

To overcome the abovementioned limitations, we propose a new rank-constrained matrix representation model with a hypergraph structure. In contrast to previously described low-rank matrix representations, this method directly handles the nonconvex decomposition model to recover the clean data from observation data simultaneously corrupted by sparse error and noise, which seeks the desired rank representation within a dictionary and separates the sparse error. The desired rank representation can be obtained by explicitly restricting the upper bound of the rank range. An alternative projection algorithm is proposed to seek the desired rank representation within a dictionary and separate the sparse error component from the corrupted data matrix. Bilateral random projections (BRPs) [18, 23] are adopted to obtain a low-rank approximation of the matrix in the iteration

procedure, which avoids the expensive computation seen in SVD. Furthermore, with the aim of utilizing the complex high-order correlations between samples, a hypergraph is constructed by grouping highly related samples into one hyperedge [21, 22]. The final clustering results are obtained by spectral decomposition of the hypergraph Laplacian matrix.

Different from [15, 16], our model produces an approximate representation Z of a matrix X in the presence of both noise G and sparse error S with the upper bound constraint of rank (Z). However, LRR [15, 16] assumes $X = XZ + S$ and decomposes X into XZ and S with the constraint of minimizing the rank (Z). Our model constrains the rank range of the coefficient matrix Z , which is valuable for subspace clustering problems, for example, face clustering and motion segmentation. Furthermore, we develop new ways to construct hyperedges and compute weights in hypergraph to better describe the local group information of each vertex, which make hypergraph clustering significant and very different from the standard Laplacian clustering. In summary, the main contributions of this research are as follows:

- (1) A rank-constrained matrix representation model is proposed to obtain the desired rank representation of the data by the rank upper bound constraint. An alternating projection algorithm is proposed to solve this model and bilateral random projections are used to seek designed rank approximation.
- (2) A hypergraph model is introduced to capture the complex and higher order relationships between data, in order to further improve the performance of subspace clustering.

2. Rank-Constrained Matrix Representation

Assume a set of data vectors $X = [x_1, x_2, \dots, x_n]$ in R^d are drawn from a union of k subspaces $\{S_i\}_{i=1, \dots, k}$ with unknown dimensions r_i . The objective is to cluster each sample into its underlying subspace. In real applications, the data are often simultaneously contaminated by both noise and error, and a fraction of the data vectors are grossly corrupted, or even missing. Following GoDec model [18], the observed data should be represented as

$$X = L + S + G. \quad (1)$$

Some rank-related constraint should be utilized to obtain the low-rank approximation L of X . S is the sparse error and G is the noise. However, GoDec model implicitly assumes that the underlying data structure is a single low-rank subspace. Facing the subspace segmentation task, we need to extend the recovery of corrupted data from single subspace to multiple subspaces.

To better handle the mixed data lying near multiple subspaces, a more general representation model should be adopted. Data X can be represented by a linear combination of the atoms in a dictionary $D = [d_1, d_2, \dots, d_m]$; that is, $X = DZ$. The dictionary should be learned to adapt to the

property of data X , and it is necessary to select the optimal representation under the desired property:

$$X = DZ + S + G, \quad (2)$$

where $Z = [z_1, z_2, \dots, z_n]$ is the coefficient matrix, z_i is the representation of x_i within the dictionary D , G is the noise, and S is the sparse error.

Due to the explicitly noisy corruption, the exact low-rank and sparse decomposition of LRR model [15, 16] may not exist. At the same time, the rank range of the representation cannot be directly constrained in the LRR model. However, in some cases, the range of the rank of the representation needs to be explicitly controlled according to the prior knowledge of the problem, so we need to look for a rank range-constrained representation Z . To handle these issues, we propose the rank-constrained representation of a matrix $X \in R^{d \times n}$ in noise case; that is,

$$\begin{aligned} \min_{Z, D, S, G} \quad & \|G\|_F^2 + \lambda \|S\|_{2,1} \\ \text{s.t.} \quad & X = DZ + S + G, \quad \text{rank}(Z) \leq r, \end{aligned} \quad (3)$$

where $G \in R^{d \times n}$ is the noise component, $S \in R^{d \times n}$ is the sparse error component, $D \in R^{d \times m}$ is the dictionary, and $Z \in R^{m \times n}$ is the low-rank representation. $\text{rank}(Z)$ is defined as the rank of matrix Z and r is the desired rank range. D will be learned adaptively to better represent X . $\|S\|_{2,1}$ is called the $l_{2,1}$ norm, which is defined as the sum of l_2 norm of the column of matrix S ; λ is the regularization parameter that balances the weight of the noise and sparse error components.

We use the matrix S to separate the sample specific corruptions (and outliers), which indicates the phenomenon that a fraction of the samples (i.e., columns of matrix X) are far away from the subspaces. Thus the $l_{2,1}$ norm is used to encourage the columns of S to be zero and separate the error of some specific samples. It is mentioned that the proper norm for the matrix S should be chosen according to the corruption type. Taking the element-wise sparse error for example, $\|S\|_1$ is an advisable constraint to separate the error component.

In (3), the optimal solution Z^* may not be a block diagonal due to the degeneration of sparse error and noise. However, it still serves as an affinity matrix, and spectral clustering algorithms are used on Z^* to obtain the final clustering results. For simplicity, we call model (3) the rank-constrained matrix representation (RMR) model, which is intrinsically different from the LRR model in the following:

- (1) The RMR model produces an approximated low-rank representation of a general matrix X upon dictionary D in the presence of noise G with the upper bound constraint of $\text{rank}(Z)$. However, LRR assumes that $X = DZ + S$ (where S is sparse error) and exactly decomposes X into DZ and S with the constraint of minimizing the nuclear norm $\|Z\|_*$. Furthermore, the dictionary D is adaptively learned in our model, which is good to represent data X .
- (2) LRR minimizes the convex surrogate of the rank constraint, that is, the nuclear norm of Z . Although

convex relaxation can simplify the optimization procedure, the solution is still a local optimum. However, our RMR model constrains the rank range of Z and directly addresses the nonconvex model. Constraint $\text{rank}(Z) \leq r$ is utilized to obtain the desired representation, and the prior knowledge of the problem can also be utilized to set the rank range r .

3. Solving the RMR Model

RMR model has multiple optimization variables which are Z , D , S , and G , respectively. This section proposes an optimization algorithm to solve this multiple variables model.

First, we replace the variable G with $X - DZ - S$ in accordance with the equality constraint (2) and thus objective (3) can be rewritten as a relaxed version:

$$\begin{aligned} \min_{Z, S, D} \quad & \|X - DZ - S\|_F^2 + \lambda \|S\|_{2,1} \\ \text{s.t.} \quad & \text{rank}(Z) \leq r. \end{aligned} \quad (4)$$

Then, we propose an iterative algorithm to solve (4), that is, how to estimate the low-rank term DZ and the sparse term S from X . Alternating minimization of multiple variables provides a useful framework for the derivation of iterative optimization algorithms. Z and S are two unknown matrix variables in (4). The optimization of (4) can be solved by alternately solving the following two subproblems until convergence. For the t th iteration,

$$Z^{t+1} = \arg \min_{\text{rank}(Z) \leq r} \frac{1}{2} \|X - DZ - S^t\|_F^2, \quad (5)$$

$$S^{t+1} = \arg \min_S \frac{1}{2} \|X - DZ^{t+1} - S\|_F^2 + \lambda \|S\|_{2,1}.$$

The first subproblem refers to the low-rank matrix representation of X , and it can be reformulated as finding the matrix Z with rank upper bound r to minimize the Euclidean distance of $X - DZ - S^{t-1}$:

$$Z^{t+1} = \arg \min_Z \frac{1}{2} \|X - DZ - S^t\|_F^2 + I_{C(r)}(Z), \quad (6)$$

where $I_{C(r)}(Z)$ is the indicator function of a set $C(r)$, defined as

$$\begin{aligned} & I_{C(r)}(Z) \\ &= \begin{cases} 0, & Z \in C(r) = \{Z \in R^{m \times n} \mid \text{rank}(Z) \leq r\}, \\ \infty, & Z \notin C(r). \end{cases} \end{aligned} \quad (7)$$

Here, we adopt the accelerated proximal gradient method [24] to solve (6) and the iteration formula is listed below:

$$\begin{aligned} Z^{t+1/2} &= Z^t - \mu D^T (DZ^t + S^t - X), \\ Z^{t+1} &= P_{I_{C(r)}}(Z^{t+1/2}). \end{aligned} \quad (8)$$

In (8), the first formula is the gradient descent operation and the second formula is the projection operator

$P_{I_{(Cr)}}(Z^{t+1/2})$, which means finding the rank- r approximation of $Z^{t+1/2}$. We adopt the bilateral random projections (BRPs) to obtain the rank- r approximation fast, as in [18, 23]. Given two random matrices, $M_1 \in R^{n \times r}$ and $M_2 \in R^{m \times r}$ ($r < n$), the rank- r bilateral random projections of the data matrix $Z \in R^{m \times n}$ ($m \geq n$) are computed as $Y_1 = ZM_1$ and $Y_2 = Z^T M_2$. Then, we can get

$$Z^{t+1} = Y_1 (M_2^T Y_1)^{-1} Y_2^T. \quad (9)$$

We can see that the inverse of $r \times r$ matrix and three matrix multiplications need to be calculated. $2mnr$ floating-point multiplications are required to obtain Y_1 and Y_2 , and $r^2(2n + r) + mnr$ multiplications are needed to obtain Z^{t+1} . Thus, the computational cost is much less than SVD-based approximation with complexity of $m^2n + n^3$ [23]. Nesterov's accelerated strategy is also adopted to further improve the convergence speed. Please refer to [24] for more details of Nesterov's accelerated strategy.

For the second subproblem, there is a closed solution. Let $T = X - DZ^{t+1}$; S can be updated by column-wise soft threshold shrinkage of T , which is just a linear complexity operation.

For the third subproblem, it corresponds to the dictionary updating. In order to reduce the computational complexity, we adopt the gradient descent method to update D , which is defined as follows:

$$D^{t+1} = D^t - \gamma (D^t Z^{t+1} + S^{t+1} - X) Z^{t+1T}, \quad (10)$$

where γ is the iterative step-size. In terms of the self-expressive property [15, 16], each data point drawing from a union of subspaces can be effectively reconstructed by a linear combination of other data points lying in the space. Namely, the sample set X can be adopted as the dictionary to represent the sample set themselves. Thus, we initialize the D^0 as the sample set X ; that is, $D^0 = X$.

The solving of three subproblems should be repeated until the stopping criterion ($relErr1 < \varepsilon_1$ and $relErr2 < \varepsilon_2$) is met or the maximum number, t_{\max} , of iterations is reached. $\varepsilon_1, \varepsilon_2$ are a small tolerance constant. $relErr1$ and $relErr2$ measure the reconstruction error and the relative variation of variables Z, S between the $t + 1$ th and t th iteration, respectively, which are calculated as follows:

$$\begin{aligned} relErr1 &= \frac{\|X - X^t Z^t - S^t\|_F^2}{\|X\|_F^2}, \\ relErr2 &= \max \left(\frac{\|Z^{t+1} - Z^t\|_F^2}{\|X\|_F^2}, \frac{\|S^{t+1} - S^t\|_F^2}{\|X\|_F^2}, \right. \\ &\quad \left. \frac{\|D^{t+1} - D^t\|_F^2}{\|X\|_F^2} \right). \end{aligned} \quad (11)$$

The complete optimization algorithm for RMR is summarized in Algorithm 1.

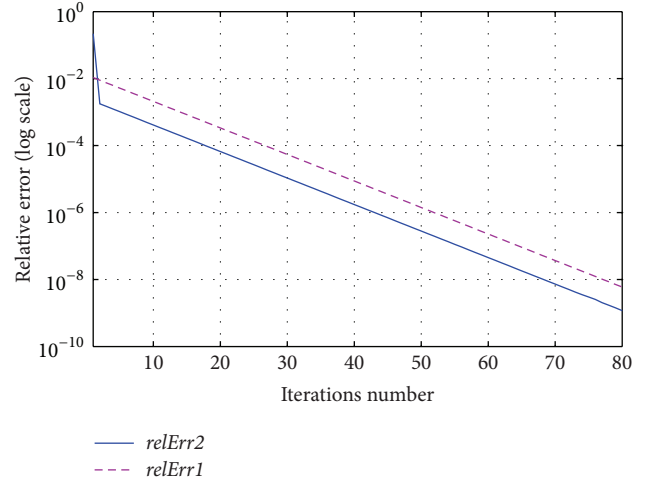


FIGURE 1: Relative error $relErr1$ and $relErr2$ versus iterations number. The parameter of RMR is set to be $\varepsilon = 1.0 \times 10^{-8}$.

Due to the alternative iteration between two variables, it is difficult to give the theoretical proof of the convergence of Algorithm 1. Nevertheless, we find that it converges asymptotically in our experiments. Figure 1 plots the curve of relative reconstruction error $relErr1$ and the relative variation $relErr2$ of variables Z, S (log scale) versus iterations number in face clustering experiment upon the Extended Yale Face Database B. The relative errors $relErr1$ and $relErr2$ both decay rapidly with the number of iterations, which indicates the convergence of our optimization algorithm.

4. Hypergraph-Based Subspace Clustering

When we obtain the rank-constrained representation of the data by the RMR model, we can calculate the similarity matrix W for spectral clustering as in [15, 16] by

$$w_{i,j} = \frac{|Z_{i,j}| + |Z_{j,i}|}{2}. \quad (12)$$

The popular way is to construct an undirected pairwise graph with weight $w_{i,j}$ assigned to the edge linking x_i and x_j , and spectral clustering is performed on the Laplacian matrix of the graph. Reference [15] stated that the large coefficients in the low-rank representation matrix usually cluster in groups. Our experiments also demonstrated the same conclusion in the proposed method. As shown in Figure 6, the large coefficients of matrix Z cluster in groups along the main diagonal in face clustering experiment upon the Extended Yale Face Database B. The i th datum has close relationships with the whole set of prominent data in its rank-constrained reconstruction, and the relation among them is more high order than pairwise. It is implied that local group information is useful for clustering. As a result of the multivariate relation being broken into many pairwise edge connections, the conventional pairwise graph is insufficient to capture the high-order relationship. Group information among the data ought to be utilized for clustering besides the pairwise relationship between two samples.

Input: Data matrix X , desired rank r , parameter λ
Output: Z^* , S^*
Initialization: $t = 0$, $D^0 = X$, $t_{\max} = 260$, $\varepsilon_1 = 1 \times 10^{-11}$, $\varepsilon_2 = 1 \times 10^{-11}$, $\tau = 10^{-3}$
While! ($relErr1 < \varepsilon_1$ and $relErr1 < \varepsilon_2$) and $t < t_{\max}$ **do**
Step 1. low-rank part update
 $U^0 = Z^k, V^0 = Z^k$
For $k = 1, 2, \dots$ **do**
 $U^{k+1/2} = V^k - \mu D^{tT} (D^t V^k + S^t - X)$
 $U^{k+1} = Y_1 (M_2^T Y_1)^{-1} Y_2^T$, wherein $Y_1 = U^{k+1/2} M_1, Y_2 = U^{k+1/2T} M_2$
 $t_{k+1} = \frac{1 + \sqrt{4t_k^2 + 1}}{2}$
 $\lambda_k = 1 + \frac{t_k - 1}{t_{k+1}}$
If $\|U^{k+1} - U^k\|_F^2 < \tau$, **return** $Z^{t+1} = U^{k+1}$;
 $V^{k+1} = U^{k+1} + \lambda_k (U^{k+1} - U^k)$;
End For
Step 2. sparse part update
set $T = X - D^t Z^{t+1}$;
shrink each column of S^{t+1} with soft-threshold

$$S_i^{t+1} = \begin{cases} \frac{\|t_i\|_2 - \lambda}{\|t_i\|_2}, & \text{if } \lambda < \|t_i\|_2 \\ 0, & \text{otherwise.} \end{cases}$$
where t_i is the i th column of T .
Step 3. dictionary update
Equation (10);
 $t = t + 1$;
End While

ALGORITHM 1: Rank-constrained matrix representation.

In contrast to pairwise graph, a hypergraph is a generalization of a graph, where each edge (called hyperedges) can connect more than two vertices. Vertices with similar characteristics can all be enclosed by a hyperedge, so high-order information of the data besides pairwise information can be effectively captured, which may be very useful for subspace clustering tasks.

In this section, we propose a method for constructing the so-called RMR-HyperGraph, in which the vertices involve all the samples and the hyperedge e_i associated with each vertex X_i describes its rank-constrained driven reconstruction. For each data point X_i , RMR-HyperGraph seeks the K most relevant neighbors in its rank-constrained representation Z_i to form a hyperedge, so that the data points in hyperedge e_i have strong dependency. The weight of each hyperedge e_i is computed to reveal the homogeneity degree of all the data points in the hyperedge. The task of subspace clustering is formulated as a problem of hypergraph partition.

4.1. Hypergraph Preliminaries. Hypergraph $G = (V, E, w)$ is formed by the vertex set V , the hyperedge set E , and the hyperedge weight vector w . Here, each hyperedge $e_i \in E$ is a subset of V and is assigned a positive weight $w(e_i)$. A $|V| \times |E|$

incidence matrix H denotes the relationship between the vertex and the hyperedge, defined as

$$h(v, e) = \begin{cases} 1, & \text{if } v \in e, \\ 0, & \text{otherwise.} \end{cases} \quad (13)$$

Based on H , the vertex degree of each vertex $v_i \in V$ and the edge degree of hyperedge $e_i \in E$ can be calculated as

$$\begin{aligned} d(v_i) &= \sum_{e_j \in E} w(e_j) h(v, e_j), \\ \delta(e_i) &= \sum_{v \in V} h(v, e_i). \end{aligned} \quad (14)$$

Let D_v and D_e denote the diagonal matrices containing the vertex and hyperedge degrees, respectively, and let W denote the diagonal matrix containing the weights of hyperedges.

4.2. Hyperedge Construction and Weight Computation. Hypergraph is actually a generalization of pairwise graph, and its key issues are how to build hyperedges and compute their weights. Most previous works have adopted K -NN searching to generate the hyperedge, whereby each sample and its K nearest neighbors are taken as a hyperedge. The

method is simple, but the fixed number of neighbors (the size of hyperedge) is not adaptive to local data distribution of each data point. Using the similarity matrix W defined in (11), we define a hypergraph G with adaptive neighbors. The K neighbors of each vertex v_i are identified as the samples whose coefficients are ranked as the first K which is largest in the i th column of W (denoted by W_i). Meanwhile, the number of neighborhoods K is adaptively selected for each vertex v_i , according to the following rule:

$$\begin{aligned} K &= \arg \min_k \\ &\text{s.t. } \frac{\|\max(W_i, k)\|_2^2}{\|W_i\|_2^2} \geq 0.8, \end{aligned} \quad (15)$$

where vector $\max(W_i, k)$ is comprised of the first k largest elements of W_i . This rule means that the energy of first k largest coefficients corresponding to the K -nearest neighbor samples is at least 80% of the energy of W_i . Then, each vertex (data sample) and its K -nearest neighbors are linked as a hyperedge.

As in [25, 26], we also relax the incidence matrix H of hypergraph G with a soft way defined as

$$h(v_i, e_j) = \begin{cases} w_{i,j}, & \text{if } v_i \in e_j, \\ 0, & \text{otherwise.} \end{cases} \quad (16)$$

According to this assignment, v_i is ‘‘partly’’ assigned to e_j based on the similarity $w_{i,j}$ between v_i and v_j , if v_i belongs to e_j . It presents not only the local grouping information, but also the importance of a vertex belonging to a hyperedge. In this way, the correlation between vertices is more accurately described.

The hyperedge weight $w(e_i)$ is computed as the sum of the pairwise similarities within the hyperedge:

$$w(e_i) = \sum_{v_j \in e_i, j < i} w_{i,j}. \quad (17)$$

Based on this definition, the ‘‘compact’’ hyperedge (local group) is assigned by a higher weight.

4.3. Hypergraph Spectral Decomposition for Subspace Clustering. Based on the constructed hypergraph, a hypergraph Laplacian matrix is constructed to find the spectrum signature of the dataset for subspace clustering based on hypergraph spectral analysis [27]. The principal idea is to perform spectral decomposition on the Laplacian matrix of the hypergraph model to obtain its eigenvectors and the eigenvalues. Hypergraph Laplacian matrices are also computed as

$$L = I - D_v^{-1/2} H W D_e^{-1} H^T D_v^{-1/2}, \quad (18)$$

where D_v , D_e , and W denote the diagonal matrices of the vertex degrees, the hyperedge degrees, and the hyperedge weights, respectively. Similarly, the problem of hypergraph partition can be relaxed into a generalized eigenvalue decomposition of the hypergraph Laplacian matrix. The hypergraph-based subspace clustering algorithmic procedure can be summarized in Algorithm 2.

5. Experimental Results

In the experiments, we test the proposed method on face clustering and motion segmentation problems and compare it with some state-of-the-art algorithms, including SSC [14], LRR [15, 16], and GPCA [9]. To further investigate the effectiveness of hypergraph, we implement two versions of spectral clustering algorithms with RMR: RMR-Graph and RMR-HyperGraph. Both of them use the RMR model to obtain the low-rank representation. RMR-HyperGraph uses the hypergraph for clustering, according to Algorithm 2, while RMR-Graph uses a pairwise graph for clustering, just like the SSC and LRR methods. Experimental results of our model, including accuracy and running time, are presented and compared against other competing methods. The parameters of all other competing algorithms are selected to have optimal results.

5.1. Simulated Data. The first experiment tests on the simulated data. We construct seven independent subspaces whose bases are generated by random rotation of the base of the previous one. Each subspace has a dimension of 5 and 30 data vectors with a dimension of 70, which are randomly sampled from each subspace. In order to simulate the noisy and error-corrupted data, each sample point x is added by a small Gaussian noise with zero mean and variance $\rho_n * \|x\|^2$. Meanwhile, a certain percentage, ρ_e , of points are also selected to add a large Gaussian noise with variance $0.3 * \|x\|^2$; these points can be regarded as outliers deviating from the original subspace.

In order to test the stability of various algorithms, we conduct two group experiments which, respectively, perform clustering under various percentages of corruption and noise intensity. First, ρ_e is fixed and set as 0.2, while ρ_n is varied from 0% to 60%. Second, ρ_e is fixed and set as 20%, and ρ_n is varied from 0 to 0.6. After performing the RMR decomposition to obtain the coefficient matrix, we use RMR-Graph and RMR-HyperGraph to segment the data into seven clusters and compare the segmentation accuracy with LRR and SSC. The regularization parameters λ are set as $\lambda_{\text{SSC}} = 0.01$, $\lambda_{\text{LRR}} = 0.12$, and $\lambda_{\text{RMR}} = 0.23$ for SSC, LRR, and RMR, respectively, which are both tuned to achieve the best performance by cross-validation. The parameter rank r is set at 35 for our method, and the experiments are repeated 10 times to obtain the mean accuracy.

As shown in Figures 2 and 3, RMR is superior to both LRR and SSC, especially when the percentage of the outliers and the noise variances increases. The performance of RMR-HyperGraph is also comparable to or better than RMR-Graph. These results demonstrate that the application of hypergraph clustering techniques improves robustness to noise and corruption for data clustering problem.

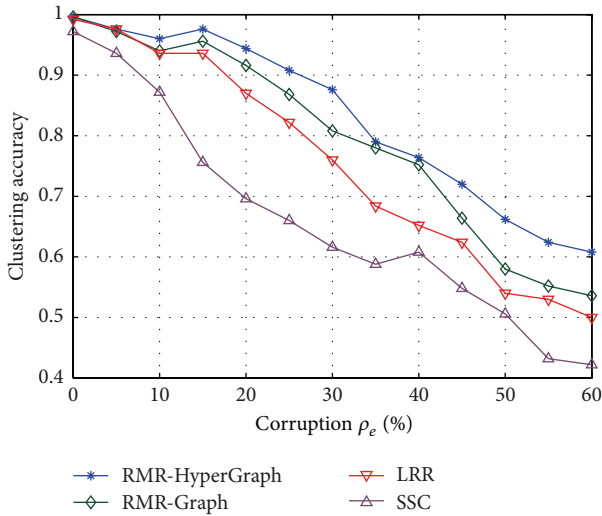
5.2. Data-Hopkins 155 Motion Dataset. Hopkins 155 motion dataset is an extensive benchmark for testing the feature-based motion segmentation algorithms [27]. It contains 155 sequences with two and three motions (each motion corresponds to a subspace). These sequences can roughly be

Input: data matrix X , number of classes C , rank r of coefficient matrix Z

- (1) Obtain the rank-constrained matrix representation via optimization Algorithm 1.
- (2) Construct a K -nearest neighbors hypergraph by using the r -rank representation to define the hyperedge and H matrix of the hypergraph.
- (3) Compute the hypergraph Laplacian matrix via (18).
- (4) Spectral decomposition of hypergraph Laplacian matrix and Take the first C eigenvectors with non-zero eigenvalues as the embedded representation.

Output: Use a K -means clustering algorithm on the eigenspace to partition the vertices of the graph into C clusters.

ALGORITHM 2: Hypergraph-based subspace clustering.


 FIGURE 2: Subspace clustering accuracy of LRR and RMR versus the percentage corruption ρ_e .

divided into three categories: checkerboard sequences, traffic sequences, and “other” (articulated/nonrigid sequences). Some sample images with superpositioned trajectory points are shown in Figure 4. For each sequence, the trajectories are extracted automatically with a tracker and outliers are manually removed. Therefore, the trajectories are only corrupted by noise but do not have missing entries or outliers. It could be considered that this database only contains slight corruptions.

The task of motion segmentation is to cluster these N trajectories tracked and extracted from the video sequence into n different groups, so that the trajectories in the same group represent a single rigid-body motion [27]. Each sequence is a sole clustering task; there are, therefore, 156 clustering tasks in total. The feature point trajectories of a single rigid-body motion lie in an affine subspace of dimension of, at most, three. Given N trajectories of n rigidly moving objects, these trajectories can be approximately regarded as lying in a union of n affine subspaces.

The tested algorithms include SSC, LRR, and RMR-HyperGraph in this experiment. For RMR, the parameters are set as $\lambda = 0.26$ and $r = 3 * n$, where n is the number of moving objects in the sequences. The parameters of the other methods are also optimally tuned. Segmentation accuracy of

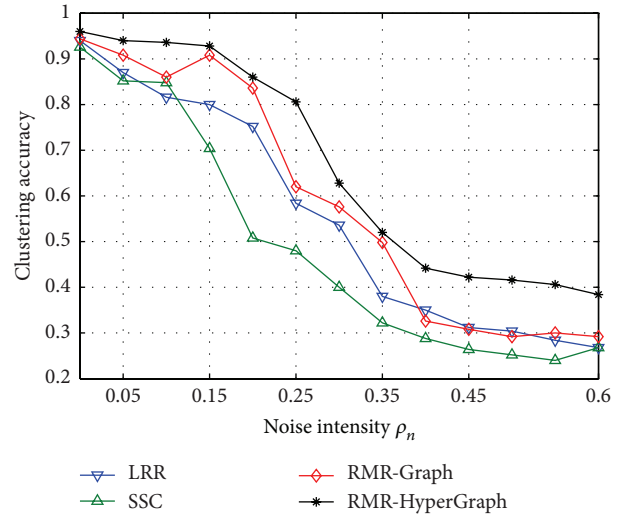

 FIGURE 3: Subspace clustering accuracy of LRR, SSC, and RMR versus the noise variance ρ_n .

TABLE 1: Segmentation accuracy (%) of sequences with two motions.

Algorithm	GPCA	SSC	LRR	RMR-Graph	RMR-HyperGraph
Mean	95.41	97.42	97.55	97.48	97.59
Median	99.62	100.00	100.00	100.00	100.00
Std	7.39	5.75	5.49	5.51	5.47

the different algorithms, including the mean, median, and standard deviation (std), is listed in Table 1 (two motion sequences) and Table 2 (three motion sequences). In this experiment, segmentation accuracies of RMR-HyperGraph and LRR are superior to GPCA and SSC method. And the accuracies of RMR and LRR algorithms are very similar. To some extent, low-rank representation is good enough to recover the subspace structure in this case of approximately clean trajectory data. The utility of the rank range constraint and hypergraph model is not to be exerted fully.

5.3. Extended Yale Face Database B. The Extended Yale Face Database B [28] consists of 640 frontal face images in 38 classes. For fair comparison, we select the first 10 classes in these experiments as in [15, 16]. Each class contains 64 images



FIGURE 4: Sample images from the Hopkins 155 database.



FIGURE 5: Some samples taken from Extended Yale Face Database B.

TABLE 2: Segmentation accuracy (%) of sequences with three motions.

Algorithm	GPCA	SSC	LRR	RMR-Graph	RMR-HyperGraph
Mean	71.34	94.71	95.41	95.40	95.87
Median	71.62	99.60	100.00	99.80	100.00
Std	11.79	7.21	6.38	6.36	6.35

with different illumination conditions. The image is taken in the different illumination condition (see Figure 5). Most of the data samples are corrupted by shadows and noise, making it an ideal test bed for the proposed algorithm.

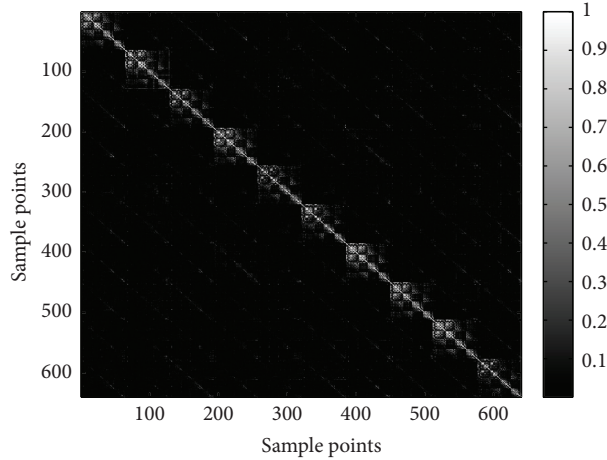
As expected for an object showing Lambertian properties, the set of all images taken under varying lighting conditions cluster in a cone of the image space, which can be approximated very well by a 9-dimensional linear subspace [20]. With the assumption that the subspace of each individual is

TABLE 3: Clustering accuracy on the Extended Yale Face Database B.

Algorithm	SSC	LRR	RMR-Graph	RMR-HyperGraph
Acc. (%)	57.66	62.53	63.67	71.46

independent, the rank is set as $r = 90$. The parameter λ is, respectively, set as $\lambda_{\text{SSC}} = 0.17$, $\lambda_{\text{LRR}} = 0.18$, and $\lambda_{\text{RMR}} = 0.24$ for SSC, LRR, and RMR algorithms to obtain the optimal performance by cross-validation.

Figure 6 displays the resulting coefficient matrix Z with large coefficients clustering in the main diagonal and small coefficients irregularly scattered. The clustering results are listed in Table 3. It shows that our algorithm significantly outperforms the others. This is because RMR can effectively recover the rank-constrained representation of a set of data vectors from corrupted data. At the same time, the application of hypergraph clustering also enhances its robustness to noise

FIGURE 6: The coefficient matrix Z of RMR for Extended Yale Face Database B.

and corruption. Figure 7 compares the decomposition examples of RMR and LRR method. The low-rank component XZ^* is expected to recover the clean sample data. We can see that our model can more effectively remove the shadow or stripe noises.

The rank upper bound r is an important parameter in the proposed model. Figure 8 reports the performance of RMR-HyperGraph with different r . It can be seen that RMR model gets approximately best performance when r is approximately 90, which is consistent with the prior knowledge that each subject has a 9-dimensional linear lighting subspace. In this case, the rank of the coefficient matrix of LRR model is about 135, which deviates largely from the prior rank range of 90. And the clustering accuracy of LRR is negatively affected by the inaccurate constraint of rank range. Our RMR model directly controls the rank range, which can address the mentioned problem of LRR and improve clustering accuracy.

In order to further test robustness to noise with various variances, a small Gaussian noise with zero mean and variance $\rho_n * \|x\|$ is added to the face images; ρ_n varies from 0.01 to 0.12. Figure 9 displays the clustering accuracy of RMR-HyperGraph, RMR-Graph, and the LRR method for different ρ_n values. We can see that RMR-HyperGraph always achieves a better performance than RMR-Graph and LRR.

5.4. AR Face Database. The AR face database contains over 4000 images corresponding to 126 subjects (70 men and 56 women) [29]. They have different facial expressions, illuminations, and occlusions. Each class contains about 26 images with resolution 55×40 . Some sample images are showed in Figure 10. In particular, the large occlusions by sunglasses or scarves make the corruption more severe than that in Extended Yale Face Database B.

We use the first 10 classes to test the proposed methods. Similarly, we set the parameters $r = 90$ and $\lambda = 0.24$ for our method empirically on this database. The parameters of the other methods are also tuned for optimal performance.

Table 4 lists the clustering results. Our algorithm is more robust for gross corruption and significantly superior to the

TABLE 4: Clustering accuracy on the AR database.

Algorithm	SSC	LRR	RMR-Graph	RMR-HyperGraph
Acc. (%)	23.66	32.53	42.37	49.23

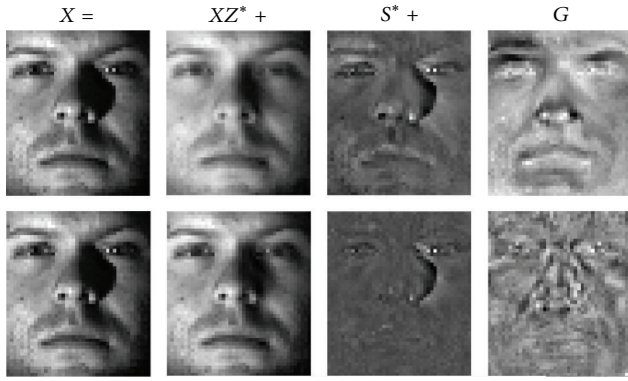
TABLE 5: Running time (in seconds) of each algorithm in the multiple face and motion datasets.

Database	SSC	LRR	RMR-HyperGraph
Hopkins 155 Motion Dataset	10934.65	2873.17	2024.28
Extended Yale Face Database B	2498.6	157.82	106.46
AR Face Database	904.2	30.98	21.13

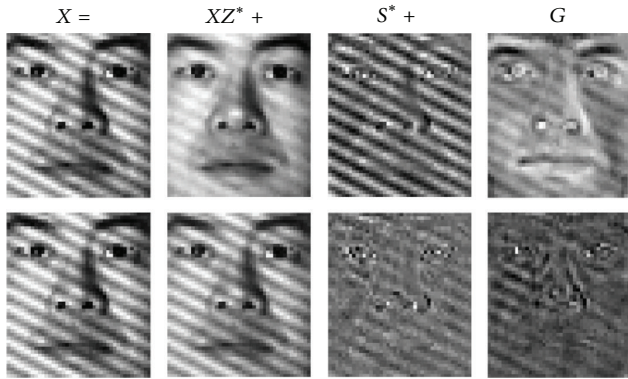
other algorithm. The decomposed components of our RMR and LRR model are shown in Figure 11. In this case of very heavy corruption, LRR fails to recover the region occluded by glasses and scarves. Nevertheless, our RMR model can still recover partial details occluded by the sunglasses or scarves, which verifies the utility of setting rank range. As shown in Figure 11(c), it is also interesting that our RMR model can alleviate the expression change and recover the normal face. Our RMR algorithm can result in robust segmentation accuracy even in the case of very serious corruption.

5.5. Running Time. We continue to analyze the computational complexity of each algorithm. The codes of SSC [14] and LRR [15, 16] are downloaded on the authors' homepage, respectively. In particular, the code of CVX version is employed to run SSC algorithm. All the algorithms are implemented in Matlab R2011b running on Windows 7, with an Intel-Core i7-2600 3.40 GHz processor and 8 GB memory. The running time (in seconds) of each algorithm in the face databases and motion datasets is listed in Table 5.

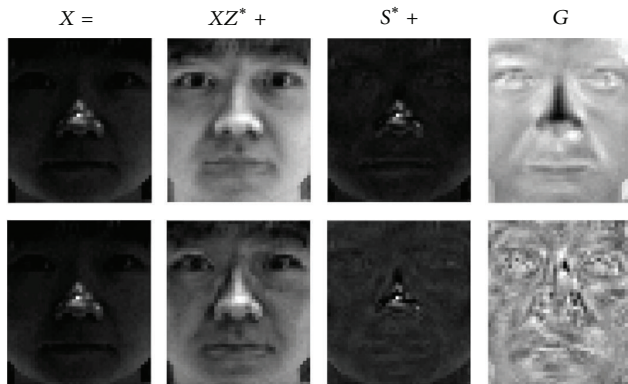
We can see that SSC is the most time-consuming method and our algorithm requires the least time. With regard to LRR



(a) Decomposition examples of the first classes



(b) Decomposition examples of the second classes



(c) Decomposition examples of the fifth classes

FIGURE 7: The decomposed components of some samples taken from Extended Yale Face Database B. The first rows in (a), (b), and (c) display the decomposition of our RMR method and the second row corresponds to the results of LRR method.

method, the low computational cost of our algorithm mainly benefits from the use of bilateral random projections (BRPs) to compute the low-rank approximation. The computational cost of BRPs is less than SVD-based approximation in LRR method.

6. Conclusion

Real-world data are frequently corrupted with both sparse error and noise. While low-rank and sparsity models have

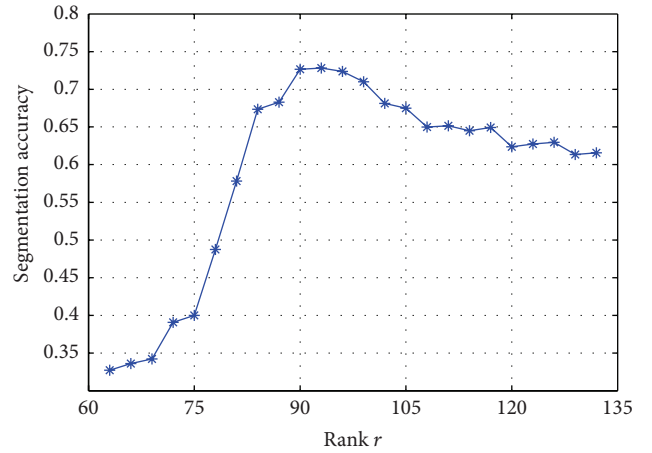


FIGURE 8: Clustering accuracy of our algorithm with different r values upon Extended Yale Face Database B.

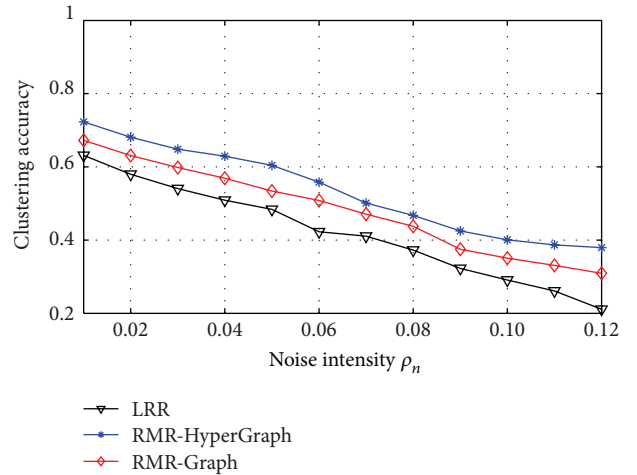


FIGURE 9: Subspace clustering accuracy with different ρ_n values upon Extended Yale Face Database B.

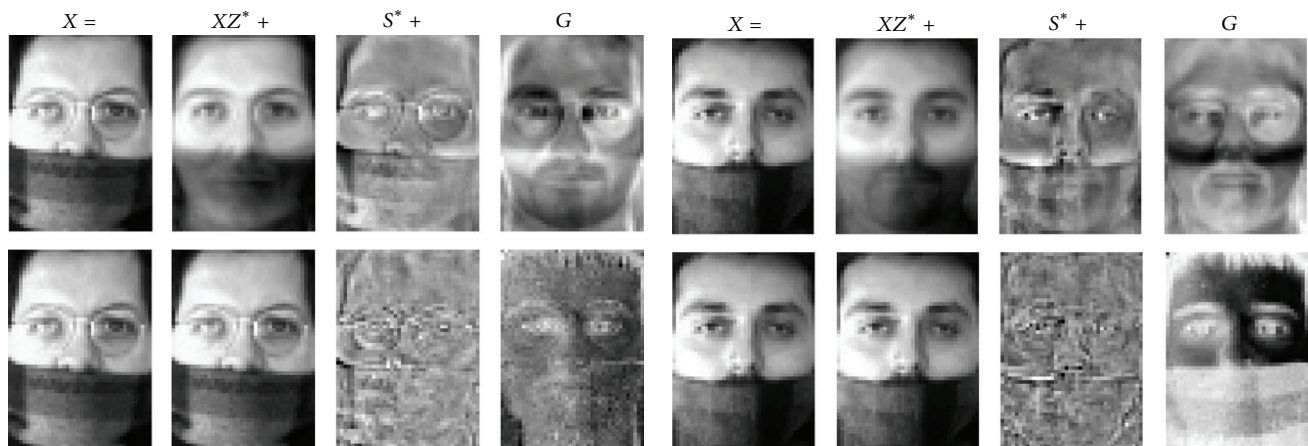
been extensively studied by the communities of computer vision and machine learning, here we propose a novel rank-constrained matrix representation model, which is able to recover a low-rank representation from the noisy and corrupted data. RMR produces an approximate representation of a matrix X in the presence of both noise G and sparse error S with the upper bound constraint of rank (Z). RMR constrains the rank range of the coefficient matrix Z , which is valuable for subspace clustering problems, for example, face clustering and motion segmentation. Meanwhile, we combine RMR with hypergraph spectral clustering, which utilizes the high-order correlations between the data points. The RMR model was experimentally tested on face clustering and motion segmentation tasks; the experimental results demonstrated the power of the proposed algorithm against the state-of-the-art algorithms.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

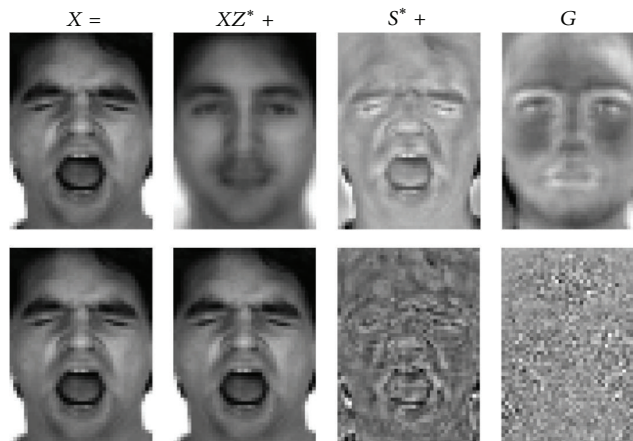


FIGURE 10: Some samples taken from Extended Yale Face Database B.



(a) Decomposition examples of the third classes

(b) Decomposition examples of the fourth classes



(c) Decomposition examples of the ninth classes

FIGURE 11: The decomposed components of some samples taken from AR. The first rows in (a), (b), and (c) display the decomposition of our RMR method and the second row corresponds to the results of LRR method.

Acknowledgments

This work was supported in part by the Natural Science Foundation of China (NSFC) under Grant 61300162, Grant 61272223, and Grant 81201161 and in part by the Natural Science Foundation of Jiangsu Province, China, under Grant BK20131003 and Grant BK2012045.

References

- [1] R. Vidal, "Subspace clustering," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 52–68, 2011.
- [2] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1031–1044, 2010.

- [3] J. Ho, M.-H. Yang, J. Lim, K.-C. Lee, and D. Kriegman, "Clustering appearances of objects under varying illumination conditions," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 1-18, IEEE, June 2003.
- [4] A. Y. Yang, J. Wright, Y. Ma, and S. S. Sastry, "Unsupervised segmentation of natural images via lossy data compression," *Computer Vision and Image Understanding*, vol. 110, no. 2, pp. 212-225, 2008.
- [5] R. Vidal, "A tutorial on subspace clustering," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 52-68, 2011.
- [6] T. E. Boult and L. G. Brown, "Factorization-based segmentation of motions," in *Proceedings of the IEEE Workshop on Visual Motion*, pp. 179-186, IEEE, Princeton, NJ, USA, October 1991.
- [7] J. P. Costeira and T. Kanade, "A multibody factorization method for independently moving objects," *International Journal of Computer Vision*, vol. 29, no. 3, pp. 159-179, 1998.
- [8] K. Kanatani, "Motion segmentation by subspace separation and model selection," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '01)*, vol. 2, pp. 586-591, Vancouver, Canada, July 2001.
- [9] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis (GPCA)," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 12, pp. 1945-1959, 2005.
- [10] Y. Ma, A. Y. Yang, H. Derksen, and R. Fossum, "Estimation of subspace arrangements with applications in modeling and segmenting mixed data," *SIAM Review*, vol. 50, no. 3, pp. 413-458, 2008.
- [11] M. E. Tipping and C. M. Bishop, "Mixtures of probabilistic principal component analyzers," *Neural Computation*, vol. 11, no. 2, pp. 443-482, 1999.
- [12] Y. Ma, H. Derksen, W. Hong, and J. Wright, "Segmentation of multivariate mixed data via lossy data coding and compression," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 9, pp. 1546-1562, 2007.
- [13] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the Association for Computing Machinery*, vol. 24, no. 6, pp. 381-395, 1981.
- [14] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '09)*, pp. 2790-2797, IEEE, Miami, Fla, USA, June 2009.
- [15] G. Liu, Z. Lin, and Y. Yu, "Robust subspace segmentation by low-rank representation," in *Proceedings of the 27th International Conference on Machine Learning (ICML '10)*, pp. 663-670, Haifa, Israel, June 2010.
- [16] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 171-184, 2013.
- [17] P. Sprechmann, A. M. Bronstein, and G. Sapiro, "Learning efficient sparse and low rank models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1821-1833, 2015.
- [18] T. Zhou and D. Tao, "GoDec: randomized low-rank & sparse matrix decomposition in noisy case," in *Proceedings of the 28th International Conference on Machine Learning (ICML '11)*, pp. 33-40, July 2011.
- [19] P. Favaro, R. Vidal, and A. Ravichandran, "A closed form solution to robust subspace estimation and clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 1801-1807, Providence, RI, USA, June 2011.
- [20] R. Basri and D. W. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, pp. 218-233, 2003.
- [21] D. Zhou, J. Huang, and B. Scholkopf, "Learning with hypergraphs: clustering, classification, and embedding," in *Advances in Neural Information Processing Systems (NIPS) 19*, pp. 1601-1608, MIT Press, 2007.
- [22] S. Agarwal, J. Lim, L. Zelnik-Manor, P. Perona, D. Kriegman, and S. Belongie, "Beyond pairwise clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 2, pp. 838-845, IEEE, San Diego, Calif, USA, June 2005.
- [23] T. Zhou and D. Tao, "Bilateral random projections," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT '12)*, pp. 1286-1290, Cambridge, Mass, USA, July 2012.
- [24] K.-C. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems," *Pacific Journal of Optimization*, vol. 6, no. 3, pp. 615-640, 2010.
- [25] Q. Liu, Y. Huang, and D. N. Metaxas, "Hypergraph with sampling for image retrieval," *Pattern Recognition*, vol. 44, no. 10-11, pp. 2255-2262, 2011.
- [26] Y. Huang, Q. Liu, S. Zhang, and D. N. Metaxas, "Image retrieval via probabilistic hypergraph ranking," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 3376-3383, San Francisco, Calif, USA, June 2010.
- [27] R. Tron and R. Vidal, "A benchmark for the comparison of 3-D motion segmentation algorithms," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1-8, IEEE, Minneapolis, Minn, USA, June 2007.
- [28] K.-C. Lee, J. Ho, and D. J. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 684-698, 2005.
- [29] A. M. Martinez and R. Benavente, "The AR face database," CVC Technical Report 24, 1998.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

