

Structured General and Specific Multi-view Subspace Clustering

Wencheng Zhu^{a,b,c}, Jiwen Lu^{a,b,c,*}, Jie Zhou^{a,b,c}

^a*Department of Automation, Tsinghua University, Beijing, 100084, China*

^b*State Key Lab of Intelligent Technologies and Systems, Beijing, 100084, China*

^c*Beijing National Research Center for Information Science and Technology, 100084, China*

Abstract

In this paper, we propose a structured general and specific multi-view subspace clustering method for image clustering. Unlike most existing multi-view subspace clustering methods which harness the shared cluster structure to preserve the consistence between different views or utilize the diversity regularization to exploit the complementary information from different views, our method learns the structured general and specific representation matrices to obtain the common and specific characteristics of different views with structure consistence and diversity regularization. The general representation matrix guarantees the consistence between different views and the specific representation matrices indicate the diversity among different views. Hence, our method can well exploit the common structure and diversity information of multi-view data. Specifically, the proposed framework can be applied into many existing multi-view subspace clustering methods. Moreover, we develop an efficient and effective optimization approach to solve the objective function of which the time and convergence analyses are also provided. Experimental results on four benchmark datasets are presented to show the effectiveness of proposed method.

Keywords: Subspace clustering, multi-view learning, structure consistence, diversity.

*Corresponding author.

Email address: lujiwen@tsinghua.edu.cn (Jiwen Lu)

1. Introduction

Subspace clustering has been an important topic in computer vision for many years due to the extensive growth of applications such as image and motion segmentation [1, 2], face clustering [3, 4], and image representation [5, 6]. Generally, the objective of subspace clustering is to segment data points drawn from multiple low-dimensional subspaces into different clusters [7]. Numerous subspace clustering methods have been proposed in recent years such as sparse subspace clustering (SSC) [1], low-rank representation (LRR) [8], least squares regression (LSR) [2], and structured sparse subspace clustering (S3C) [9]. While these subspace clustering methods have achieved encouraging performance on single-view data, it remains a challenge to deal with multi-view data as both specific and generic information coexists in multi-view data.

In computer vision, the requirement to exploit the comprehensive information from multiple distinct features arises as the feature descriptors are easy to obtain and various multi-view datasets are created. Motivated by this, several multi-view subspace clustering methods have been proposed in recent years [10, 11]. Essentially, multi-view subspace clustering methods exploit intrinsic properties, i.e., complementarity and consistency, among different views to jointly enhance the generalization ability of learning models. Representative algorithms in multi-view subspace learning include multi-view low-rank sparse subspace clustering (MLRSSC) [12], diversity-induced multi-view subspace clustering (DiMSC) [13], exclusivity-consistency regularized multi-view subspace clustering (ECMSC) [3], and consistent and specific multi-view subspace clustering (CSMSC) [14]. These multi-view methods have greatly improved performance than single-view methods. However, the consistency based methods fail to utilize the diversity among different views and the complementary based methods insufficiently preserve the common cluster structure by enforcing the affinity matrices to be different. Hence, these methods cannot well utilize the consistency and diversity characteristics among different views.

We propose a novel structured multi-view subspace clustering approach to learn the general and specific self-representation matrices for image clustering in this paper. Figure 1 illustrates the basic idea of our proposed approach. The general self-

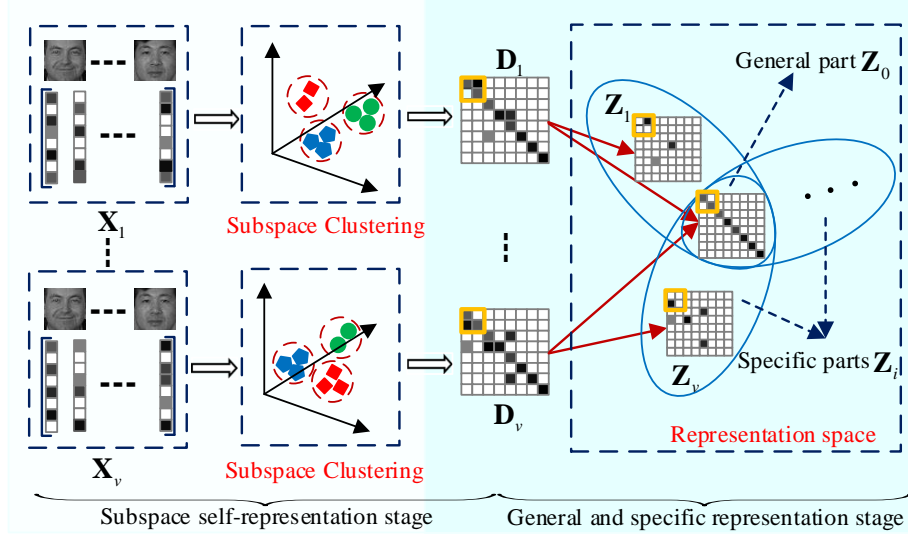


Figure 1: The basic idea of our proposed multi-view subspace clustering approach. For the i^{th} view data matrix \mathbf{X}_i , i.e., $i = 1, \dots, v$, the affinity matrix \mathbf{D}_i is obtained by subspace self-representation. Then, the general representation matrix \mathbf{Z}_0 and specific representation matrices \mathbf{Z}_i are learned from the affinity matrices $\mathbf{D}_i = \mathbf{Z}_0 + \mathbf{Z}_i$ with the consistence and diversity constraints. The general representation matrix \mathbf{Z}_0 preserves the original similarity of data and the specific representation matrices \mathbf{Z}_i hold the diversity of data. Respectively, we apply the structure consistence on general representation matrix and diversity constraints on specific representation matrices. As shown in the orange square boxes, the connections between different \mathbf{Z}_i and \mathbf{Z}_0 are weakened.

representation matrix \mathbf{Z}_0 is a matrix shared by affinity matrices \mathbf{D}_i , which follows the consistence principle to maintain the general property that when data points are close in original feature space, their corresponding representations in a new embedding space should be close. While the specific representation matrices \mathbf{Z}_i hold specific property that the matrices \mathbf{Z}_i are different to each other and complementary to the general self-representation matrix for representing their specific similarities of corresponding spaces. Figure 1 shows that \mathbf{D}_1 and \mathbf{D}_v share \mathbf{Z}_0 , \mathbf{Z}_1 and \mathbf{Z}_v are complementary to \mathbf{Z}_0 to build \mathbf{D}_1 and \mathbf{D}_v . Generally, the data point can be approximately reconstructed by the weighted sum of neighboring points, and the weights which are invariant to linear transformation of data reflect the inherent geometric and similarity properties. Particularly, the same weights are preserved in different embedding spaces [15]. Thus,

the weight matrices or similarity matrices in different feature spaces share the intrinsic and common similarity which is entitled as general representation matrix in this paper. And the learnt general representation matrix also keeps the similarity relationship of the original multi-view data. Subspace self-representation builds the affinity matrix to measure the similarity between data points when the sparse or low-rank constraint is imposed on regularization term [1, 8]. For multi-view subspace self-representations, they have the structural shared and general property and also the specific relationship with the diversity information. Motivated by these findings, our proposed method learns the structured general and specific representation matrices to represent the intrinsic similarity relationship and explore the diversity of different views simultaneously. Experimental results are presented to demonstrate the effectiveness of our proposed method.

The contributions of our proposed approach are summarized as follows:

1. We propose a structured multi-view subspace clustering approach to learn the structural general representation matrix and specific representation matrices.
2. The general representation matrix keeps the similarity relationship of data and the specific representation matrices hold diversity between different matrices.
3. We present an effective optimization algorithm to solve the proposed objective function.
4. We conduct experiments on four benchmark datasets and the experimental results demonstrate the effectiveness of our proposed method.

2. Related Work

In this section, we briefly review two related works: 1) multi-view learning, and 2) subspace clustering.

2.1. Multi-view Learning

Existing multi-view learning methods [16, 17] can be mainly categorized into three classes: co-training based, co-regularization based, and margin-consistency based [18]. The co-training methods alternately update the parameters of models on different views

70 and classify the unlabeled data with confident labeled ones. One of the most representative algorithm is the co-training multi-view spectral clustering method [19]. Unlike co-training methods which have independent parameters for each views, co-regularization methods impose the same regularization term on different models. Typical method in this category is co-regularized multi-view spectral clustering [20]. Margin-consistency
75 methods learn the margin variables of each view to fulfill the consistence principle. For example, RMSC [21] pursues a shared low-rank transition matrix which is incorporated into the standard Markov method. FOLS [22] seeks shared and private factorization when the data are transformed into a latent space while finds the the dimensionality of this latent space together. However, FLOS is formulated as a complicated and
80 non-convex problem. LRCS [23] obtains a common mapping for different views from view-specific projections, but whether the mapping can be factorized needs to be considered further. Otherwise, LRCS needs the dimensions of views to be equal. These methods learn the shared and private latent representations or projections, respectively. However, they fail to uncover the subspace structure of different views. Our proposed
85 method exploits subspace self-representations to pursue the general and specific structures of different views and the structured general and specific representation matrices can be obtained.

2.2. Subspace Clustering

Subspace clustering has attracted much attention in recent years [24, 25, 26], be-
90 cause subspace clustering methods can obtain ideal affinity matrices of which the elements from different subspaces are zero. Such affinity matrices are informative for spectral clustering. SSC [1] exploits the sparsity among data points in the same subspace. However, SSC has difficulties in segmenting data when the sparse constraint is too severe [2]. Low-Rank Representation (LRR) [8] pursues the low-rank representa-
95 tion instead of sparse representation with the goal of clustering samples into respective subspaces and removing outliers. Least squares regression (LSR) [2] finds that the affine matrix can be block diagonal without sparse or low-rank constraint and obtains affine matrix via least squares regression. However, LSR is sensitive to noise. S3C [9] formulates the sparse subspace learning and spectral clustering into a unified frame-

Table 1: The notations of variables used in this paper

Notations	Explanation
$\mathbf{X}_i \in \mathbb{R}^{d_i \times n}$	Data matrix for i^{th} view
$\mathbf{E}_i \in \mathbb{R}^{d_i \times n}$	Reconstruction error matrix for i^{th} view
$\mathbf{D}_i \in \mathbb{R}^{n \times n}$	Self-representation matrix for i^{th} view
$\mathbf{Z}_i \in \mathbb{R}^{n \times n}$	Specific representation matrix for i^{th} view
$\mathbf{Z}_0 \in \mathbb{R}^{n \times n}$	General representation matrix
$\mathbf{Y}_i \in \mathbb{R}^{d_i \times n}$	the Lagrange multipliers for i^{th} view
$\mathbf{H}_i, \mathbf{G}_i \in \mathbb{R}^{n \times n}$	the Lagrange multipliers for i^{th} view
$\mathbf{J}_i \in \mathbb{R}^{n \times n}$	Relaxation variables for i^{th} view
$\mathbf{L} \in \mathbb{R}^{n \times n}$	Laplacian matrix
$[\mathbf{A}]_{:,k}$	k^{th} column of matrix A
$\ \cdot\ _*, tr(\cdot)$	Nuclear norm and Trace operator
n, v	Number of samples and views
d_i	Dimension of i^{th} view data matrix

work to learn the affinity matrix and coefficient matrix together. However, S3C is hard to approach to a point and needs more iterations. To deal with the multi-view data, Many multi-view subspace clustering methods have been developed. MLRSSC [12] conducts sparse and low-rank subspace clustering on each view and then incorporates the affinity matrices together to obtain the common cluster. DiMSC [13] and ECM-SC [3] utilize the diversity between different views and pursue the cluster structure with complementarity information. CSMSC [14] exploits the consistent and specific information of multi-view affinity matrices. However, the constraints have little meaning because CSMSC cannot guarantee that the learned matrix holds the consistent principle and preserves the similarity information of original data as well as the diversity of specific matrices. These multi-view clustering methods all have achieved good performance. But they cannot exploit the intrinsic characteristics because the affinity matrices have structural general and specific properties.

3. Proposed Approach

In this section, we elaborate the details of our proposed approach. The notations
115 are summarized in Table 1.

As shown in Figure 1, we propose a structured general and specific multi-view
subspace clustering method. First, we learned the affinity matrices \mathbf{D}_i in the subspace
self-representation stage by using low-rank subspace clustering (LRR). The affinity
matrices capture the low-rank structure of each view. Then, we obtained the general
120 representation matrix \mathbf{Z}_0 and specific representation matrices \mathbf{Z}_i with the consistence
and diversity constraints. Finally, we conducted the spectral clustering [27] by using
 $\mathbf{Z} = \sum_i (\mathbf{Z}_i + \mathbf{Z}_0)$.

3.1. Multi-view Subspace Clustering

Let $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$ be the data matrix, where d and n are the dimen-
sionality of the feature space and the number of data points respectively. Subspace clus-
tering aims to obtain a self-representation matrix to better reconstruct the data points
themselves and can be formulated as the following problem:

$$\min_{\mathbf{D}, \mathbf{E}} \|\mathbf{E}\|_l + \lambda \|\mathbf{D}\|_k \quad s.t. \quad \mathbf{X} = \mathbf{X}\mathbf{D} + \mathbf{E}, \quad (1)$$

where $\|\cdot\|_l$ and $\|\cdot\|_k$ are two norms, $\mathbf{D} \in \mathbb{R}^{n \times n}$ and $\mathbf{E} \in \mathbb{R}^{d \times n}$ are the self-representation
and reconstruction error matrices. Due to the outliers and sample-specific corruptions
among data, the following robust learning model is selected,

$$\min_{\mathbf{D}, \mathbf{E}} \|\mathbf{E}\|_{2,1} + \lambda \|\mathbf{D}\|_* \quad s.t. \quad \mathbf{X} = \mathbf{X}\mathbf{D} + \mathbf{E}, \quad (2)$$

where $\ell_{2,1}$ -norm is denoted as $\|\mathbf{A}\|_{2,1} = \sum_i \|[\mathbf{A}]_{:,i}\|_2$. It can be inferred that the
125 columns of matrix tend to be zero when $\ell_{2,1}$ -norm is minimized. The above optimiza-
tion problem in (2) is a modified version of LRR and the theoretical and experimental
results have been given to guarantee the robustness to outliers and sample-specific cor-
ruptions in [8].

Having solved (2), we obtain the affinity matrix $\mathbf{S} = (|\mathbf{D}| + |\mathbf{D}^T|)/2$ on which the
spectral clustering is to be conducted .

$$\min_{\mathbf{F}} tr(\mathbf{F}^T(\mathbf{O} - \mathbf{S})\mathbf{F}) \quad s.t. \quad \mathbf{F}^T\mathbf{F} = \mathbf{I}, \quad (3)$$

where \mathbf{F} is a cluster structure matrix and \mathbf{O} is a diagonal matrix with the elements $o_{ii} = \sum_j s_{ij}$. By solving the minimization problem (3), we obtain the result of spectral clustering eventually.

Naturally, the extension of single-view subspace clustering to the multi-view case arrives at the following problem:

$$\min_{\mathbf{D}_i, \mathbf{E}_i} \sum_{i=1}^v \|\mathbf{E}_i\|_{2,1} + \lambda_1 \sum_{i=1}^v \|\mathbf{D}_i\|_* \quad s.t. \quad \mathbf{X}_i = \mathbf{X}_i \mathbf{D}_i + \mathbf{E}_i, \quad i = 1, \dots, v. \quad (4)$$

Specially, the subspace self-representation matrices hold the common cluster structure and we utilize the general matrix \mathbf{Z}_0 to implement the structure consistence across different views. Simultaneously, the subspace self-representation matrices also contain the individual information expressed as the specific matrices \mathbf{Z}_i to indicate diversity of each view. We exploit the constraint $\mathbf{D}_i = \mathbf{Z}_0 + \mathbf{Z}_i$ to model the connection between each view. Considering that the differences of similarity relationships between different views exist but are not extensive, we impose the ℓ_1 -norm on \mathbf{Z}_i . As we know, the ℓ_1 -norm tends to force the elements of matrix to be zero. Hence, a few elements of \mathbf{Z}_i are nonzero with specific information. The constraints lead to the problem:

$$\begin{aligned} \min_{\mathbf{D}_i, \mathbf{E}_i, \mathbf{Z}_i, \mathbf{Z}_0} \quad & \sum_{i=1}^v \|\mathbf{E}_i\|_{2,1} + \lambda_1 \sum_{i=1}^v (\|\mathbf{D}_i\|_* + \|\mathbf{Z}_i\|_1) \\ s.t. \quad & \mathbf{X}_i = \mathbf{X}_i \mathbf{D}_i + \mathbf{E}_i, \mathbf{D}_i = \mathbf{Z}_0 + \mathbf{Z}_i, i = 1, \dots, v. \end{aligned} \quad (5)$$

The performance of our model largely hinges on the learnt general representation matrix \mathbf{Z}_0 as well as the specific matrices \mathbf{Z}_i . The model in (5) has two drawbacks: one is that the model uses the element-wise loss with no structural loss incorporated. As we know the general representation matrix also follows the consistence principle and holds similar relationships in original data. The other is that the specific representation matrices should be different as far as possible to represent the peculiarity of each view. Hence, we introduce two constraints including structure consistence and diversity regularization in our model.

3.2. Structure Consistence

In multi-view learning, the assumption that different views share the intrinsic property to guarantee the consistence has been demonstrated in many literatures [28, 29]. As for our model, the consistence is ensured via the general representation matrix \mathbf{Z}_0 .

Following the definition in [30], the close data in the feature space have close representations in the representation space that is $\|\mathbf{x}_i - \mathbf{x}_j\|_2 \rightarrow 0 \Rightarrow \|\mathbf{z}_i - \mathbf{z}_j\|_2 \rightarrow 0$ where \mathbf{x}_i and \mathbf{x}_j are the data points in the feature space with the corresponding representations \mathbf{z}_i and \mathbf{z}_j . The general matrix \mathbf{Z}_0 preserves the consistence and should conform to the definition.

Suppose that $\mathbf{W} \in \mathbb{R}^{n \times n}$ measures the similarity of data points. \mathbf{z}_i and \mathbf{z}_j are i^{th} and j^{th} columns in general representation matrix \mathbf{Z}_0 , we formulate the definition as:

$$\Theta(\mathbf{Z}_0) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n w_{ij} \|\mathbf{z}_i - \mathbf{z}_j\|_2^2 = \text{tr}(\mathbf{Z}_0 \mathbf{L} \mathbf{Z}_0^T), \quad (6)$$

where $\mathbf{L} = \mathbf{R} - \mathbf{W}$ is a Laplacian matrix, \mathbf{R} is a diagonal matrix with the elements $r_{ii} = \sum_j w_{ij}$. When $\Theta(\mathbf{Z}_0)$ is minimized, the large w_{ij} leads to the close representations of \mathbf{z}_i and \mathbf{z}_j . The equation in (6) implies that the learnt general representation matrix \mathbf{Z}_0 accords with the similarity relationship in original feature space.

Define the weight vector $\mathbf{q} = [q^1, \dots, q^i, \dots, q^v] \in \mathbb{R}^v$ where v is the total number of views and q^i reflects the positive weight of \mathbf{W}^i . We use the constraint $\sum_i q^i = 1$ to limit the weight vector. The matrix \mathbf{W} can be built via the weighted sum of similarity matrices \mathbf{W}^i . Hence, we construct the similarity matrix \mathbf{W} by $\mathbf{W} = \sum_i q^i \mathbf{W}^i$ where the similarity matrix \mathbf{W}^i is computed using kernel function like gaussian kernel function. If there is no prior information about the weight vector, each view is of equal weight and we simply set $\mathbf{q} = [1/v, \dots, 1/v]$.

The structure consistence constraint on the general representation matrix guarantees the learned matrix preserves similarity information of each original feature space.

3.3. Diversity Regularization

Multi-view data maintain the respective and exclusive properties which are expressed as specific matrices other than shared and common information. Obviously, the specific representation matrices should be different with each other to highlight the diversity. Moreover, the constraint $\mathbf{D}_i = \mathbf{Z}_0 + \mathbf{Z}_i$ indicates that \mathbf{Z}_0 and \mathbf{Z}_i should be different and complementary. Otherwise, a more better \mathbf{Z}_0 can be learned to decrease the value of $\|\mathbf{Z}_i\|_1$.

To implement that each specific representation has weak connection with each other as well as the general representation, we employ a diversity regularization. For matrices \mathbf{Z}_u and \mathbf{Z}_v , the diversity regularization is formulated as:

$$\Theta(\mathbf{Z}_u, \mathbf{Z}_v) = \text{tr}(\mathbf{Z}_u^T \mathbf{Z}_v) = \sum_{i=1}^n \sum_{j=1}^n (\mathbf{Z}_u)_{ij} (\mathbf{Z}_v)_{ij}, \quad (7)$$

where $(\mathbf{Z}_v)_{ij}$ is the element of \mathbf{Z}_v . From the equation in (7), we can deduce that a large value of $(\mathbf{Z}_v)_{ij}$ results in a small value of $(\mathbf{Z}_u)_{ij}$ when $\Theta(\mathbf{Z}_u, \mathbf{Z}_v)$ is minimized. Hence, we can use the regularization to evaluate the diversity of different matrices. We formulate the diversity regularization in our model as

$$\sum_{i=1}^v \Theta(\mathbf{Z}_i, \mathbf{Z}_0) + \sum_{i=1}^v \sum_{i \neq j} \Theta(\mathbf{Z}_i, \mathbf{Z}_j) = \sum_{i=1}^v \text{tr}(\mathbf{Z}_i^T \mathbf{Z}_0) + \sum_{i=1}^v \sum_{i \neq j} \text{tr}(\mathbf{Z}_i^T \mathbf{Z}_j). \quad (8)$$

To enhance the consistency and diversity of subspace self-representation matrices, we integrate the terms together and obtain the final objective function:

$$\begin{aligned} \min_{\mathbf{E}_i, \mathbf{D}_i, \mathbf{Z}_i, \mathbf{Z}_0} & \sum_{i=1}^v \|\mathbf{E}_i\|_{2,1} + \lambda_1 \sum_{i=1}^v (\|\mathbf{D}_i\|_* + \|\mathbf{Z}_i\|_1) + \lambda_2 \text{tr}(\mathbf{Z}_0 \mathbf{L} \mathbf{Z}_0^T) \\ & + \lambda_3 \sum_{i=1}^v \text{tr}(\mathbf{Z}_i^T \mathbf{Z}_0) + \lambda_3 \sum_{i=1}^v \sum_{i \neq j} \text{tr}(\mathbf{Z}_i^T \mathbf{Z}_j) \\ \text{s.t. } & \mathbf{E}_i = \mathbf{X}_i - \mathbf{X}_i \mathbf{D}_i, \mathbf{D}_i = \mathbf{Z}_i + \mathbf{Z}_0, i = 1, \dots, v \end{aligned} \quad (9)$$

where λ_1 , λ_2 and λ_3 are the positive parameters which balance the normalization term, structure consistence term and diversity term.

4. Optimization

We propose an optimization method to approximatively solve the objective function in (9) and apply the Augmented Lagrange Multiplier (ALM) [31, 32] to iteratively update all the variables \mathbf{D}_i , \mathbf{Z}_i , \mathbf{E}_i and \mathbf{Z}_0 , $i = 1, 2, \dots, v$. We also study the model complexity following the optimization approach.

4.1. Proposed Approach

First, we introduce auxiliary variables $\mathbf{D}_i = \mathbf{J}_i$ to make the objective function become separable and can be easily solved. Then, we use the Augmented Lagrange

Multiplier (ALM) [31] to obtain the following augmented Lagrangian function:

$$\begin{aligned} \min_{\mathbf{E}_i, \mathbf{D}_i, \mathbf{Z}_i, \mathbf{Z}_0, \mathbf{J}_i} & \sum_{i=1}^v \|\mathbf{E}_i\|_{2,1} + \lambda_1 \sum_{i=1}^v (\|\mathbf{J}_i\|_* + \|\mathbf{Z}_i\|_1) + \lambda_2 \text{tr}(\mathbf{Z}_0 \mathbf{L} \mathbf{Z}_0^T) \\ & + \lambda_3 \sum_{i=1}^v \sum_{i \neq j} \text{tr}(\mathbf{Z}_i^T \mathbf{Z}_j) + \lambda_3 \sum_{i=1}^v \text{tr}(\mathbf{Z}_i^T \mathbf{Z}_0) + \sum_{i=1}^v \Phi(\mathbf{G}_i, \mathbf{D}_i - \mathbf{J}_i) \\ & + \sum_{i=1}^v \Phi(\mathbf{Y}_i, \mathbf{X}_i - \mathbf{X}_i \mathbf{D}_i - \mathbf{E}_i) + \sum_{i=1}^v \Phi(\mathbf{H}_i, \mathbf{Z}_i + \mathbf{Z}_0 - \mathbf{D}_i), \end{aligned} \quad (10)$$

where $\Phi(\mathbf{H}, \mathbf{D})$ is deemed as $\frac{\mu}{2} \|\mathbf{D}\|_F^2 + \langle \mathbf{H}, \mathbf{D} \rangle$, μ is a penalty parameter, and \mathbf{Y}_i , \mathbf{G}_i and \mathbf{H}_i are lagrange multipliers. The objective function in (10) is hard to solve and we iteratively update one variable with the other variables fixed. The detailed procedure is described as follows.

1. \mathbf{J}_i -subproblem: We update the variable \mathbf{J}_i by fixing the other variables and dropping the unconcerned terms. Then, we have the following subproblem with respect to the variable \mathbf{J}_i :

$$\begin{aligned} \min_{\mathbf{J}_i} & \lambda_1 \|\mathbf{J}_i\|_* + \Phi(\mathbf{G}_i, \mathbf{D}_i - \mathbf{J}_i) \\ & = \lambda_1 \|\mathbf{J}_i\|_* + \frac{\mu}{2} \|\mathbf{J}_i - \mathbf{D}_i - \mathbf{G}_i/\mu\|_F^2. \end{aligned} \quad (11)$$

We solve the subproblem in (11) by using the Singular Value Thresholding (SVT) [33, 34] and get the closed-form solution as

$$\mathbf{J}_i = \mathcal{S}_{\lambda_1/\mu}[\mathbf{D}_i + \mathbf{G}_i/\mu], \quad (12)$$

where $\mathcal{S}_\tau[\cdot]$ is the shrinkage thresholding operator [35] which is defined by

$$\begin{aligned} \mathcal{S}_\mu[\mathbf{M}] &= \mathbf{U} \mathcal{H}_\mu[\boldsymbol{\Sigma}] \mathbf{V}^T \\ \mathcal{H}_\mu[\boldsymbol{\Sigma}] &= \max(0, \boldsymbol{\Sigma} - \mu) + \min(0, \boldsymbol{\Sigma} + \mu), \end{aligned} \quad (13)$$

$\mathbf{M} = \mathbf{U} \boldsymbol{\Sigma} \mathbf{V}^T$ is the Singular Value Decomposition of \mathbf{M} .

2. \mathbf{E}_i -subproblem: Similar to the subproblem \mathbf{J}_i , the subproblem \mathbf{E}_i is settled by keeping the variables \mathbf{D}_i and \mathbf{Y}_i unchanged and omitting the unrelated ones. The optimization problem with respect to \mathbf{E}_i is formulated as:

$$\begin{aligned} \min_{\mathbf{E}_i} & \|\mathbf{E}_i\|_{2,1} + \Phi(\mathbf{Y}_i, \mathbf{X}_i - \mathbf{X}_i \mathbf{D}_i - \mathbf{E}_i) \\ & = \|\mathbf{E}_i\|_{2,1} + \mu/2 \|\mathbf{E}_i - \mathbf{X}_i + \mathbf{X}_i \mathbf{D}_i - \mathbf{Y}_i/\mu\|_F^2. \end{aligned} \quad (14)$$

Referring to the **Lemma 3.3** in [36], we have the optimal solution of (14):

$$[\mathbf{E}_i]_{:,k} = \begin{cases} \frac{\|[\mathbf{Q}_i]_{:,k}\|_2^{-1/\mu}}{\|[\mathbf{Q}_i]_{:,k}\|_2} [\mathbf{Q}_i]_{:,k} & \text{if } \|[\mathbf{Q}_i]_{:,k}\|_2 > 1/\mu \\ 0, & \text{else.} \end{cases} \quad (15)$$

$\mathbf{Q}_i = \mathbf{X}_i - \mathbf{X}_i \mathbf{D}_i + \mathbf{Y}_i/\mu$ and $[\mathbf{Q}_i]_{:,k}$ is the k^{th} column of \mathbf{Q}_i . The proof of (14) can be found in [8, 37].

3. \mathbf{D}_i -subproblem: By ignoring the irrelevant items and fixing the other variables, the optimization problem concerned with \mathbf{D}_i can be rewritten as:

$$\begin{aligned} \min_{\mathbf{D}_i} & \Phi(\mathbf{G}_i, \mathbf{D}_i - \mathbf{J}_i) + \Phi(\mathbf{H}_i, \mathbf{Z}_i + \mathbf{Z}_0 - \mathbf{D}_i) \\ & + \Phi(\mathbf{Y}_i, \mathbf{X}_i - \mathbf{X}_i \mathbf{D}_i - \mathbf{E}_i). \end{aligned} \quad (16)$$

The subproblem in (16) has the closed-form solution. We set the derivative of (16) with respect to \mathbf{D}_i to zero and obtain the following solution:

$$\mathbf{D}_i = \left(\mathbf{X}_i^T \mathbf{X}_i + 2\mathbf{I} \right)^{-1} \begin{Bmatrix} \mathbf{X}_i^T (\mathbf{X}_i + \mathbf{Y}_i/\mu - \mathbf{E}_i) + \mathbf{Z}_i \\ + \mathbf{Z}_0 + \mathbf{H}_i/\mu + \mathbf{J}_i - \mathbf{G}_i/\mu \end{Bmatrix}. \quad (17)$$

4. \mathbf{Z}_i -subproblem: According to the above subproblems, we can rewrite the optimization problem about \mathbf{Z}_i as:

$$\begin{aligned} \min_{\mathbf{Z}_i} & \lambda_1 \|\mathbf{Z}_i\|_1 + \Phi(\mathbf{H}_i, \mathbf{Z}_i + \mathbf{Z}_0 - \mathbf{D}_i) \\ & + \lambda_3 \text{tr}(\mathbf{Z}_i^T \mathbf{Z}_0) + \lambda_3 \sum_{i \neq j} \text{tr}(\mathbf{Z}_i^T \mathbf{Z}_j). \end{aligned} \quad (18)$$

To be simple, (18) is equivalently reduced to the following problem:

$$\min_{\mathbf{Z}_i} \lambda_1 \|\mathbf{Z}_i\|_1 + \mu/2 \|\mathbf{Z}_i\|_F^2 + \text{tr}(\mathbf{Z}_i^T \mathbf{C}), \quad (19)$$

where $\mathbf{C} = \lambda_3 \left(\mathbf{Z}_0 + \sum_{i \neq j} \mathbf{Z}_j \right) + \mu (\mathbf{Z}_0 + \mathbf{H}_i/\mu - \mathbf{D}_i)$. The optimization problem in (19) can be independently solved via Singular Value Thresholding (SVT) [33, 34]. Define $z = [\mathbf{Z}_i]_{kl}$ as the element of \mathbf{Z}_i . The optimization problem in (19) is divided into the following problem:

$$\min_z \lambda_1 |z| + \mu z^2/2 + c_{ij} z, \quad (20)$$

the solution in (20) is $z = \mathcal{H}_{\lambda_1/\mu}(-c_{ij}/\mu)$. Hence, we obtain the closed-form solution of (18) as:

$$\mathbf{Z}_i = \mathcal{H}_{\lambda_1/\mu}(-\mathbf{C}/\mu). \quad (21)$$

5. \mathbf{Z}_0 -subproblem: Fixing the other variables and removing the irrelevant terms, we renew the \mathbf{Z}_0 by dealing with the problem:

$$\begin{aligned} \min_{\mathbf{Z}_0} \lambda_2 \text{tr}(\mathbf{Z}_0 \mathbf{L} \mathbf{Z}_0^T) + \lambda_3 \sum_{i=1}^v \text{tr}(\mathbf{Z}_i^T \mathbf{Z}_0) \\ + \sum_{i=1}^v \Phi(\mathbf{H}_i, \mathbf{Z}_i + \mathbf{Z}_0 - \mathbf{D}_i). \end{aligned} \quad (22)$$

Similar to the subproblem \mathbf{D}_i , we set the derivative concerned with \mathbf{Z}_0 to zero and obtain the closed-form solution to \mathbf{Z}_0 as follows:

$$\mathbf{Z}_0 = \left(\mu \sum_{i=1}^v (\mathbf{D}_i - \mathbf{H}_i/\mu - \mathbf{Z}_i) - \lambda_3 \sum_{i=1}^v \mathbf{Z}_i \right) / (2\lambda_2 \mathbf{L} + \mu v \mathbf{I}), \quad (23)$$

where \mathbf{L} is the Laplacian matrix defined in (6) and \mathbf{I} is the identity matrix.

6. Update the multipliers and μ : Finally, we need update the multipliers and μ to satisfy the equality constraints.

$$\begin{cases} \mathbf{Y}_i = \mathbf{Y}_i + \mu(\mathbf{X}_i - \mathbf{X}_i \mathbf{D}_i - \mathbf{E}_i) \\ \mathbf{H}_i = \mathbf{H}_i + \mu(\mathbf{Z}_i + \mathbf{Z}_0 - \mathbf{D}_i) \\ \mathbf{G}_i = \mathbf{G}_i + \mu(\mathbf{D}_i - \mathbf{J}_i) \\ \mu = \min(\max_\mu, \rho\mu), \end{cases} \quad (24)$$

μ and ρ are the penalty variable and the step size respectively, \max_μ is the maximum value of the parameter μ . We settle the optimization problem in (10) by iteratively updating \mathbf{J}_i , \mathbf{E}_i , \mathbf{D}_i , \mathbf{Z}_i and \mathbf{Z}_0 until the change rate of the objective function is less than a threshold. To present the detailed procedure of our optimization problem, we summary the algorithm in **Algorithm 1**.

4.2. Model Complexity

In this subsection, we analyze the complexity of our proposed method. There are six subproblems shown in Algorithm 1. The time complexities of subproblems \mathbf{Z}_i and the updated process of the multipliers and μ can be omitted in comparison with

Algorithm 1: S MSC

Input: The multi-view matrices $\mathcal{X} = \{\mathbf{X}_2, \dots, \mathbf{X}_v\}$, parameters $\lambda_1, \lambda_2, \lambda_3$
Initialize: Set $\mathbf{D}_i = \mathbf{0}, \mathbf{J}_i = \mathbf{0}, \mathbf{Z}_i = \mathbf{0}, \mathbf{Z}_0 = \mathbf{0}, \mu = 0.01, \rho = 1.2, \max_{\mu} = 10^8$
and Initialize \mathbf{E}_i with a sparse matrix.
Normalize the data matrix \mathbf{X} .
Compute the Laplacian matrix \mathbf{L} .
while *not converge* **do**
 Update \mathbf{J}_i by (12);
 Update \mathbf{E}_i by (15);
 Update \mathbf{D}_i by (17);
 Update \mathbf{Z}_i by (21);
 Update \mathbf{Z}_0 by (23);
 Update the multipliers and μ by (24);
Output: The matrices $\mathbf{J}_i, \mathbf{E}_i, \mathbf{D}_i, \mathbf{Z}_i$ and \mathbf{Z}_0 ;

the other subproblems. Since the subproblem \mathbf{J}_i utilizes the shrinkage thresholding operator [33], the time complexity for updating \mathbf{J}_i is $O(n^3)$. For subproblem \mathbf{E}_i , the time complexity is $O(n^2 d_i)$. The subproblems \mathbf{D}_i and \mathbf{Z}_0 need matrix inversion, the time complexities of these two subproblems are $O(n^2 d_i + n^3)$ and $O(n^3)$ respectively. For v different views, the total complexity of our method in each iteration is $O(n^2 d + n^3)$, where $d = d_1 + \dots + d_v$.

5. Experiments and Results

First, we describe the experimental settings including datasets, comparison methods, evaluation metrics and parameter settings. Second, we show experimental results and analyses on benchmark datasets.

5.1. Experimental Settings

5.1.1. Datasets

We conducted experiments to evaluate our multi-view subspace clustering method on four widely used datasets.

200 The Yale dataset contains 165 gray-scale images from 15 persons with different facial expressions and configurations. We extracted three features including intensity, LBP and Gabor on Yale. The dimensions of these three features are 4096, 3304 and 6750 respectively. The LBP and Gabor features were normalized to unit [13]. The MSRCV1 dataset contains 240 images and 8 classes, among which 7 classes and 210
205 images were selected including tree, building, airplane, cow, face, car and bicycle. The CENTRIST, CMT, HOG, LBP and SIFT features were extracted [10]. The ORL dataset contains 400 images and 40 different subjects. For each subject, there are 10 images with variations in lightings, expressions and facial details. The same as the Yale dataset, the intensity, LBP and Gabor features were used [13]. The Still-DB dataset is an action
210 image dataset which contains 467 images and 6 classes. The features extracted are Sift Bow, Color Sift Bow and Shape context Bow [38].

5.1.2. Comparison Methods

We compared our approach with five single-view and seven multi-view state-of-the-art clustering methods. **SPC** [39], **SSC** [1], **LRR** [8], **LSR** [2], **S3C** [9] are single-view methods and **ConcatePCA** [13], **Co-Reg SPC** [20], **DiMSC** [13], **ECMSC** [3],
215 **MLRSSC** [12], **CSMSC** [14], **RMSC** [21] are multi-view methods. We obtained the publicly available source codes provided by authors. For single-view methods, we evaluated each feature independently and represented the best results as \mathbf{SPC}_{best} , \mathbf{SSC}_{best} , \mathbf{LRR}_{best} , \mathbf{LSR}_{best} and $\mathbf{S3C}_{best}$.

220 Next, we elaborate the comparison methods: **SPC** [39] is the standard method to perform spectral clustering. **SSC** [1] is an important sparse subspace clustering method for single-view data and the affinity matrix tends to be block diagonal when the subspaces are independent. **LRR** [8] is a low-rank representation method in comparison with **SSC**. **LSR** [2] learns the compact representation instead of the sparse or low-rank representation. **S3C** [9] combines the learning of cluster structure and sparse representation into a unified framework. **ConcatePCA** [13] marked as \mathbf{PCA}_{multi} is a baseline method which concatenates all the features together. We conductd PCA to reduce the dimension of concatenated feature to 300. **SPC_{co-reg}** [20] is a multi-view subspace clustering method which assumes that the multi-view data have the common underly-

Table 2: Experimental results (mean \pm standard deviation) on the Yale dataset.

	Method	NMI	ACC	AR	F-score	Precision
Single	SPC_{best}	0.654 ± 0.006	0.628 ± 0.009	0.441 ± 0.009	0.472 ± 0.008	0.460 ± 0.008
	SSC_{best}	0.705 ± 0.012	0.681 ± 0.010	0.478 ± 0.014	0.511 ± 0.013	0.487 ± 0.014
	LRR_{best}	0.722 ± 0.010	0.706 ± 0.015	0.540 ± 0.017	0.569 ± 0.016	0.557 ± 0.017
	LSR_{best}	0.742 ± 0.013	0.728 ± 0.018	0.551 ± 0.013	0.579 ± 0.012	0.561 ± 0.011
	$S3C_{best}$	0.707 ± 0.004	0.679 ± 0.002	0.490 ± 0.005	0.522 ± 0.004	0.496 ± 0.007
Multiple	PCA_{multi}	0.641 ± 0.006	0.544 ± 0.038	0.392 ± 0.009	0.431 ± 0.008	0.415 ± 0.007
	SPC_{co-reg}	0.664 ± 0.005	0.625 ± 0.007	0.459 ± 0.007	0.494 ± 0.006	0.476 ± 0.006
	$MLRSSC$	0.678 ± 0.003	0.644 ± 0.003	0.475 ± 0.005	0.508 ± 0.005	0.487 ± 0.004
	$RMSC$	0.674 ± 0.006	0.640 ± 0.009	0.468 ± 0.009	0.502 ± 0.009	0.482 ± 0.009
	$DiMSC$	0.758 ± 0.011	0.751 ± 0.018	0.586 ± 0.017	0.612 ± 0.016	0.596 ± 0.017
	$ECMSC$	0.721 ± 0.024	0.711 ± 0.019	0.491 ± 0.003	0.525 ± 0.003	0.487 ± 0.003
	$CSMSC$	0.767 ± 0.002	0.734 ± 0.011	0.589 ± 0.001	0.615 ± 0.001	0.591 ± 0.000
Ours	$SMSC$	0.792 ± 0.025	0.765 ± 0.035	0.640 ± 0.036	0.663 ± 0.034	0.639 ± 0.032

ing clustering structure. SPC_{co-reg} learns the cluster structure with the consistency across the views. **DiMSC** [13] is also a multi-view subspace clustering method which exploits the diversity between different views. DiMSC is a representative multi-view subspace clustering method and achieves excellent clustering results. **ECMSC** [3] can be deemed as a multi-view extension of S3C and also uses the complementary information between different views. **MLRSSC** [12] learns the low-rank and sparse subspace self-representations with the common cluster structure. **CSMSC** [14] only pursues the shared and individual representation matrices with no structure information. **RM-SC** [21] constructs a shared low-rank matrix which is used as the input of the standard Markov chain.

5.1.3. Evaluation Metrics

We adopted five widely used evaluation metrics to evaluate the performance of different subspace clustering methods for quantitative comparison. The metrics contain Normalized Mutual Information (NMI), Accuracy (ACC), Adjusted Rand index (AR), F-score and Precision. For all clustering methods, the larger values of metrics are expected to achieve the better performance [10, 13].

Table 3: Experimental results (mean \pm standard deviation) on the MSRCV1 dataset.

	Method	NMI	ACC	AR	F-score	Precision
Single	SPC_{best}	0.605 ± 0.011	0.688 ± 0.014	0.507 ± 0.013	0.576 ± 0.011	0.570 ± 0.011
	SSC_{best}	0.625 ± 0.002	0.761 ± 0.002	0.534 ± 0.003	0.600 ± 0.003	0.585 ± 0.003
	LRR_{best}	0.580 ± 0.021	0.664 ± 0.018	0.456 ± 0.026	0.533 ± 0.022	0.519 ± 0.020
	LSR_{best}	0.656 ± 0.003	0.693 ± 0.024	0.562 ± 0.010	0.624 ± 0.008	0.611 ± 0.008
	$S3C_{best}$	0.626 ± 0.000	0.714 ± 0.000	0.542 ± 0.000	0.609 ± 0.000	0.582 ± 0.000
Multiple	PCA_{multi}	0.465 ± 0.005	0.602 ± 0.009	0.347 ± 0.005	0.438 ± 0.005	0.434 ± 0.004
	SPC_{co-reg}	0.741 ± 0.019	0.808 ± 0.027	0.658 ± 0.028	0.707 ± 0.024	0.691 ± 0.027
	MLRSSC	0.704 ± 0.006	0.803 ± 0.007	0.612 ± 0.009	0.667 ± 0.008	0.652 ± 0.009
	RMSC	0.570 ± 0.005	0.689 ± 0.009	0.481 ± 0.007	0.554 ± 0.006	0.548 ± 0.007
	DiMSC	0.718 ± 0.008	0.844 ± 0.005	0.668 ± 0.010	0.714 ± 0.009	0.705 ± 0.010
	ECMSC	0.758 ± 0.022	0.855 ± 0.008	0.711 ± 0.011	0.752 ± 0.010	0.749 ± 0.008
	CSMSC	0.228 ± 0.001	0.333 ± 0.000	0.076 ± 0.000	0.268 ± 0.000	0.180 ± 0.000
Ours	SMSC	0.788 ± 0.002	0.876 ± 0.001	0.738 ± 0.001	0.775 ± 0.001	0.762 ± 0.000

5.1.4. Parameter Settings

For our multi-view subspace clustering method, we learned the structural general and specific representation matrices with the regularization item, consistence item and diversity term, where the tradeoff parameters λ_1 , λ_2 and λ_3 balance corresponding terms. For fair comparisons with different subspace clustering methods, we followed the experimental settings of corresponding papers and empirically tuned the parameters from the range $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ by using the grid-search strategy. The best results were reported [13, 3, 12]. In our experiments, the parameter μ is a Lagrangian penalty variable. Referring to [16, 3, 31], we initialized μ by 0.01 and set its maximum value as 10^8 . For the parameter ρ , we set its value as 1.2. We initialized all the matrices with a zero matrix except \mathbf{E}_i with a sparse matrix and applied PCA on the concatenated feature to reduce its dimension to 300. In our learning model, we utilized no prior information about different views and set all weights as $1/v$. For computing similarity matrices, we chose the linear kernel. Moreover, we iterated all subspace clustering methods for 30 times and recorded the mean values and standard deviations of experimental results. The while loop in Algorithm 1 ends when the f-norm of matrix differences between variable states and updated variable states is less

Table 4: Experimental results (mean \pm standard deviation) on the ORL dataset.

	Method	NMI	ACC	AR	F-score	Precision
Single	SPC_{best}	0.908 ± 0.003	0.782 ± 0.009	0.721 ± 0.009	0.728 ± 0.009	0.683 ± 0.011
	SSC_{best}	0.852 ± 0.011	0.688 ± 0.039	0.578 ± 0.033	0.589 ± 0.032	0.526 ± 0.042
	LRR_{best}	0.928 ± 0.005	0.831 ± 0.016	0.776 ± 0.017	0.782 ± 0.017	0.742 ± 0.021
	LSR_{best}	0.947 ± 0.007	0.870 ± 0.015	0.827 ± 0.018	0.831 ± 0.018	0.796 ± 0.020
	$S3C_{best}$	0.927 ± 0.009	0.828 ± 0.023	0.770 ± 0.029	0.775 ± 0.028	0.735 ± 0.033
Multiple	PCA_{multi}	0.807 ± 0.015	0.647 ± 0.027	0.519 ± 0.029	0.531 ± 0.029	0.497 ± 0.029
	SPC_{co-reg}	0.870 ± 0.004	0.725 ± 0.007	0.639 ± 0.010	0.648 ± 0.010	0.602 ± 0.011
	MLRSSC	0.908 ± 0.001	0.789 ± 0.004	0.731 ± 0.004	0.737 ± 0.004	0.697 ± 0.005
	RMSC	0.925 ± 0.004	0.813 ± 0.011	0.767 ± 0.012	0.772 ± 0.012	0.731 ± 0.013
	DiMSC	0.940 ± 0.006	0.863 ± 0.013	0.815 ± 0.018	0.819 ± 0.017	0.787 ± 0.020
	ECMSC	0.947 ± 0.009	0.854 ± 0.011	0.810 ± 0.012	0.821 ± 0.015	0.783 ± 0.008
	CSMSC	0.931 ± 0.005	0.839 ± 0.010	0.787 ± 0.012	0.792 ± 0.013	0.751 ± 0.013
Ours	SMSC	0.957 ± 0.006	0.889 ± 0.015	0.855 ± 0.017	0.859 ± 0.017	0.827 ± 0.020

than 10^{-4} . The experiments ran on MATLAB R2015b with a computer having Intel(R) Core(TM) i5-6500 CPU with 3.2GHZ and 32.0GB RAM.

265 5.2. Results and Analysis

5.2.1. Comparisons with State-of-the-Arts

Tables 2-5 show the experimental results on the Yale, MSRCV1, ORL and Still-DB datasets, respectively. These tables show that our proposed method SMSC achieves the superior performance and outperforms all the other methods by five evaluation metrics
270 on the Yale, MSRCV1, ORL datasets. On the Still-DB dataset, our method obtains comparable results to the other multi-view subspace clustering methods.

Table 2 shows the experimental results on the Yale dataset. Our method outperforms all the other methods on NMI, ACC, AR, F-score, Precision and improves the baseline SPC_{best} and PCA_{multi} more than 10 %. PCA_{multi} is the worst method be-
275 cause the features are simply concatenated together and more noise will be introduced. While S3C integrates sparse subspace learning and spectral clustering into a unified framework to pursue a better cluster indicator, the method is very time-consuming and hard to converge. Thus, S3C has poor performance than LRR and LSR. SPC_{co-reg}

Table 5: Experimental results (mean \pm standard deviation) on the Still-DB dataset.

	Method	NMI	ACC	AR	F-score	Precision
Single	SPC _{best}	0.108 \pm 0.002	0.299 \pm 0.002	0.063 \pm 0.001	0.226 \pm 0.001	0.232 \pm 0.001
	SSC _{best}	0.117 \pm 0.000	0.326 \pm 0.000	0.075 \pm 0.000	0.291 \pm 0.000	0.228 \pm 0.000
	LRR _{best}	0.113 \pm 0.002	0.298 \pm 0.004	0.068 \pm 0.001	0.226 \pm 0.001	0.228 \pm 0.001
	LSR _{best}	0.117 \pm 0.004	0.314 \pm 0.005	0.070 \pm 0.002	0.232 \pm 0.002	0.228 \pm 0.001
	S3C _{best}	0.127 \pm 0.009	0.338 \pm 0.000	0.083 \pm 0.003	0.293\pm0.006	0.234 \pm 0.000
Multiple	PCA _{multi}	0.105 \pm 0.002	0.289 \pm 0.002	0.056 \pm 0.001	0.232 \pm 0.000	0.214 \pm 0.001
	SPC _{co-reg}	0.120 \pm 0.004	0.322 \pm 0.004	0.082 \pm 0.003	0.245 \pm 0.004	0.236 \pm 0.002
	MLRSSC	0.104 \pm 0.001	0.315 \pm 0.000	0.069 \pm 0.001	0.250 \pm 0.003	0.226 \pm 0.000
	RMSC	0.115 \pm 0.001	0.331 \pm 0.002	0.085 \pm 0.002	0.252 \pm 0.003	0.236 \pm 0.003
	DiMSC	0.136 \pm 0.003	0.344 \pm 0.003	0.093\pm0.002	0.258 \pm 0.002	0.245 \pm 0.002
	ECMSC	0.117 \pm 0.001	0.340 \pm 0.002	0.088 \pm 0.000	0.259 \pm 0.001	0.240 \pm 0.000
	CSMSC	0.128 \pm 0.000	0.315 \pm 0.000	0.081 \pm 0.001	0.253 \pm 0.001	0.232 \pm 0.000
Ours	SMSC	0.140\pm0.002	0.345\pm0.003	0.089 \pm 0.002	0.259 \pm 0.002	0.248\pm0.002

and MLRSSC both assume that the multi-view data have the shared affinity matrix by all views. Hence, these two methods have the comparable performances. But, SPC_{co-reg} and MLRSSC have the worse results than the other single-view methods because they hardly consider the specific information. DiMSC and ECMSC exploit the complementary information among different views and the performance is better than SPC_{co-reg} and MLRSSC. ECMSC alternatively learns the exclusive representations and the consistent indicator. However, iteratively pursuing the consistent indicator by spectral clustering is time-consuming and we show the average computational time of ECMSC in time analysis subsection. CSMSC pursues the shared and individual self-representation matrices. Due to no structure information and diversity regularization exploited in CSMSC, the learnt matrices have little clear meaning. Our method improves the performance of CSMSC by 2.5 %. RMSC first constructs the similarity matrices for each view and then pursues the shared low-rank transition matrix incorporated into the standard Markov process. However, RMSC uses little specific information between different views. The performance of RMSC is comparable to these methods utilizing the common information. Our method learns the structural general and specific representation matrices ensuring the consistency and diversity between d-

Table 6: Experimental results (mean \pm standard deviation) on the Yale dataset.

Parameters	NMI	ACC	AR	F-score	Precision
λ_1 only	0.421 \pm 0.006	0.383 \pm 0.006	0.117 \pm 0.005	0.185 \pm 0.004	0.147 \pm 0.004
λ_2 only	0.763 \pm 0.015	0.732 \pm 0.022	0.588 \pm 0.024	0.614 \pm 0.022	0.594 \pm 0.025
λ_3 only	0.767 \pm 0.019	0.745 \pm 0.028	0.601 \pm 0.028	0.626 \pm 0.027	0.610 \pm 0.027
λ_1, λ_2	0.389 \pm 0.011	0.352 \pm 0.011	0.081 \pm 0.009	0.161 \pm 0.006	0.113 \pm 0.008
λ_1, λ_3	0.209 \pm 0.005	0.207 \pm 0.004	0.004 \pm 0.002	0.095 \pm 0.003	0.064 \pm 0.001
λ_2, λ_3	0.756 \pm 0.003	0.720 \pm 0.005	0.569 \pm 0.008	0.597 \pm 0.007	0.567 \pm 0.011
$\lambda_1, \lambda_2, \lambda_3$	0.792\pm0.025	0.765\pm0.035	0.640\pm0.036	0.663\pm0.034	0.639\pm0.032

ifferent views. Additionally, our method is more efficient than ECMSC and can achieve a better performance compared with DiMSC.

Table 3 presents the experimental results on the MSRCV1 dataset. Our method also works well on this dataset evaluated by five evaluation metrics and improves the performance of PCA_{multi} more than 20 %. For single-view methods, SSC and LSR have higher performance. Generally, the performance of multi-view methods is better than single-view ones. The MSRCV1 dataset holds more shared information and the complementary information has little effect on the results. Thus, multi-view methods pursuing common cluster structure work better compared with diversity methods.

Table 4 describes the experimental results on the ORL dataset. Our method outperforms all the other methods on five evaluation metrics and improves PCA_{multi} more than 10 %. LSR has the best result than the other comparison methods except SM-SC. DiMSC and ECMSC obtain the comparable performance while SPC_{co-reg} and MLRSCC are worse than most single-view methods.

Table 5 shows the experimental results on the Still-DB dataset. Our method outperforms all the other methods on three evaluation metrics. For F-score metric, the single-view method S3C achieves the best result and we can deduce that there exists a good feature. Thus, the multi-view methods SPC_{co-reg} and MLRSSC pursuing the common structure obtain poor performance.

For single-view subspace clustering methods, they can provide the quality information about each view as every single view is assessed independently. A good result for a single-view method suggests that good features exist. From the tables, we can deduce

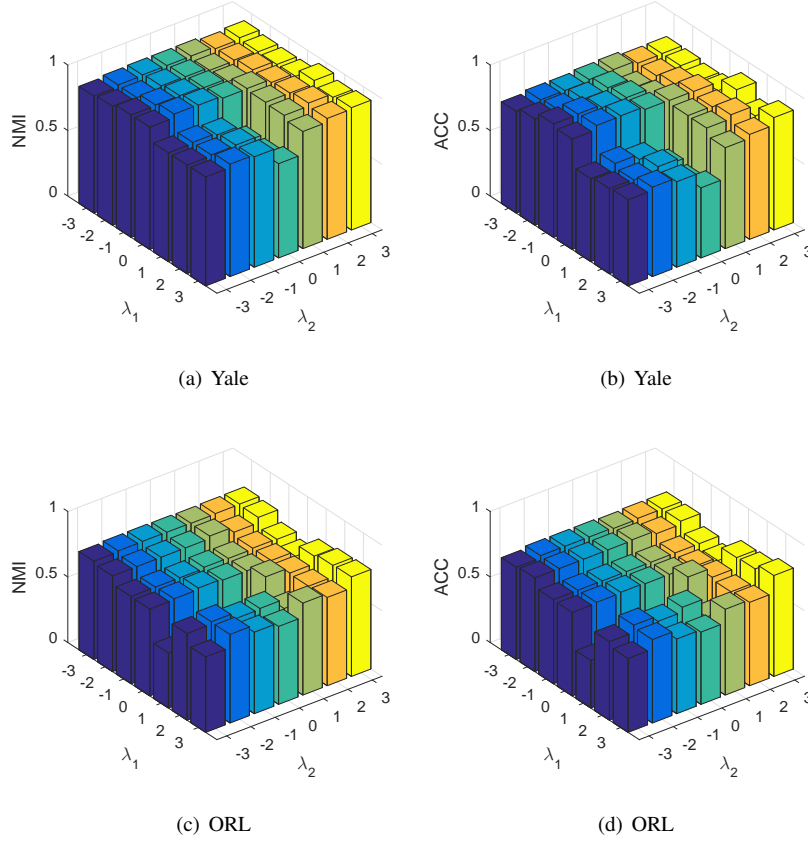


Figure 2: The results of NMI, ACC on the Yale, ORL datasets with different parameters when we set $\lambda_3 = 1$.

that the ORL and Still-DB datasets have good features. S3C is very time-consuming and hard to converge to a good local point due to the joint learning of the cluster indicator matrix and affinity matrix. Hence, S3C always has the poor performance than single-view methods in Tables 2-4. For the Still-DB dataset, S3C may arrive at a good local point. SSC and LRR have a bad performance in Table 2-4, because these methods build the assumption that data lie on a sparse or low-rank subspace. However, the data matrix may be not sparse or low-rank. LSR assumes that the data matrix has compact representations instead and this method achieves better results than most single-view methods on four datasets.

For multi-view data sharing more information between different views, our method

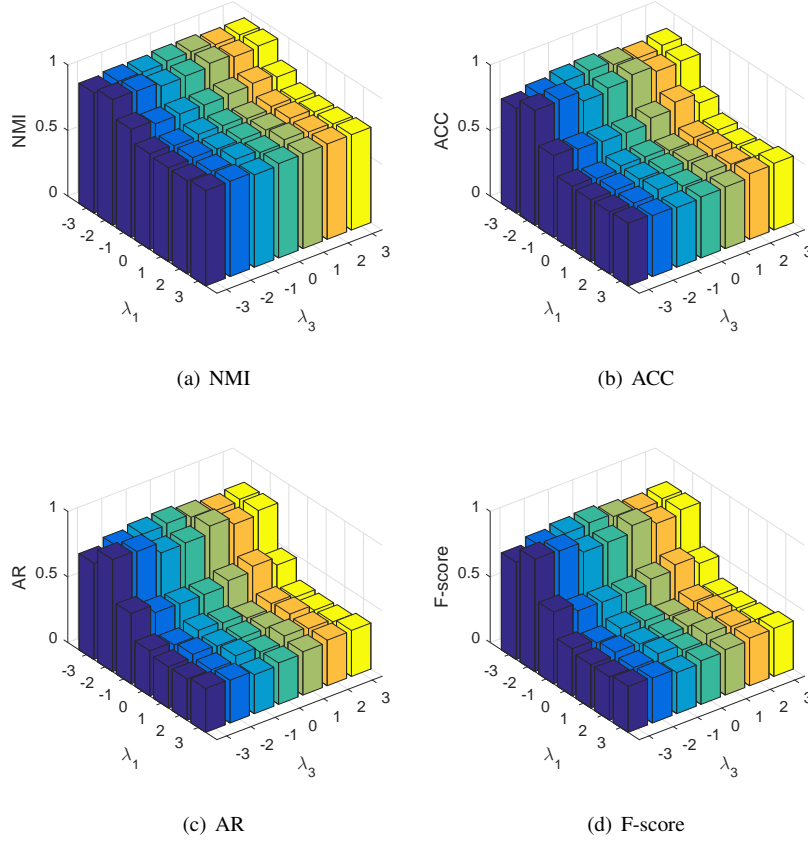


Figure 3: The results of four metrics on the ORL dataset with different parameters when we set $\lambda_2 = 1$.

obtains a general representation matrix which reflects the strong connections between data. While the multi-view data share little information, the better specific representation matrices can be constructed to represent the individual characteristics. Our model learns more individual information in such case. Hence, unlike other multi-view methods, our proposed method can handle different types of multi-view data and has good generalization capability. Generally speaking, Multi-view methods utilize multi-view information and achieve the better results. PCA_{multi} has the worst performance for all datasets. In Table 4, there exist good features and LSR outperforms MLRSSC and SPC_{co-reg} in which the common affinity matrix is pursued. MLRSSC and SPC_{co-reg} achieve good improvements on the Yale, MSRCV1 and Still-DB datasets which have

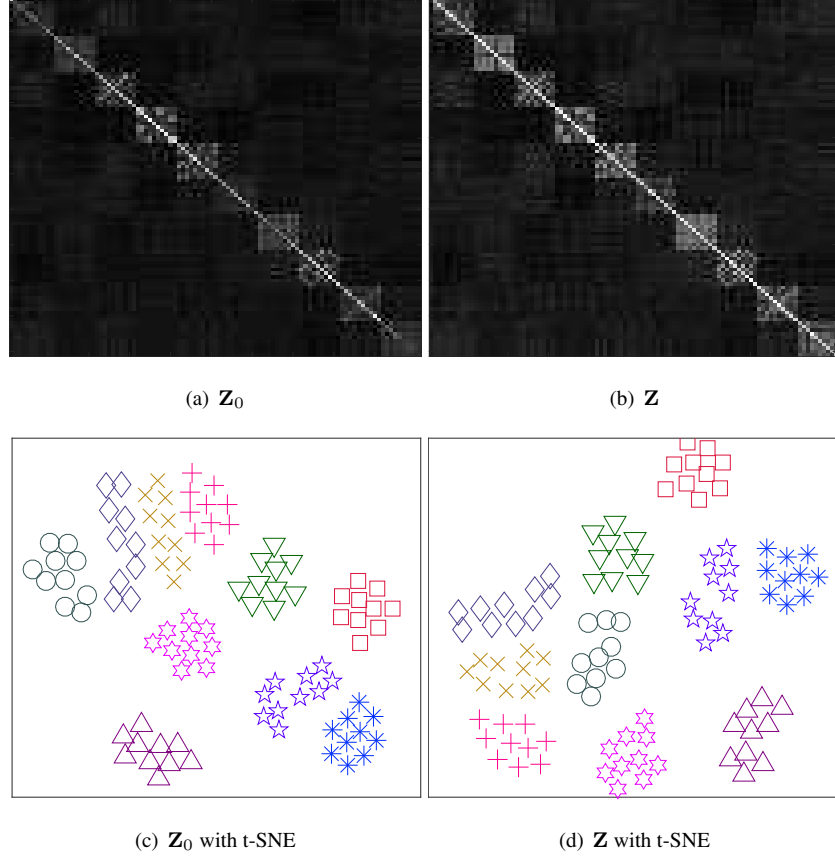


Figure 5: The representation matrices of \mathbf{Z}_0 , \mathbf{Z} and the visualizations of these two matrices with t-SNE on the ORL dataset.

ing. From the table, we can validate that both structure consistence term and diversity regularization term are important for our proposed model.

5.2.3. Parameter Analysis

360 We analyzed the sensitivity of parameters in Figures 2 and 3 (the base of logarithm is 10). In Figure 2, we show the NMI and ACC results on the Yale and ORL datasets. The subfigures (a) and (b) are the results on the Yale dataset while the subfigures (c) and (d) are the results on the ORL dataset. We see that the performances of SMSC are almost equal with different parameters λ_1 and λ_2 on the Yale and ORL datasets.

365 Hence, we can simply set $\lambda_1 = \lambda_2$ to tune parameters for convenience. In Figure 3, we

Table 7: Average computational time (seconds) of different multi-view subspace clustering methods.

	MVLRSSC	SPC _{co-reg}	DiMSC	ECMSC	CSMSC	SMSC
Yale	1.68	3.12	3.09	80.50	10.17	9.05
MSRCV1	3.36	2.27	6.69	12.56	9.59	3.91
ORL	6.51	15.84	17.20	147.00	65.68	39.28
Still-DB	7.11	3.65	11.96	12.50	53.73	9.49

present the NMI, ACC, AR and F-score results on the ORL dataset. Figure 3 shows that our method is not sensitive to λ_3 when λ_1 is fixed. We can also observe that λ_1 plays an important role in improving the performance on the Yale dataset. In our model, the parameter λ_1 controls the learning of low-rank affinity matrix \mathbf{D}_i . Hence, a suitable λ_1 tends to achieve a better performance which gives a good explanation for Figure 3.

5.2.4. Visualization

Figure 4 visualizes some clustering results on the Yale dataset. We selected 12 individuals and each individual has 8 images. Specifically, each row is supposed to present 16 images for two persons of which the first 8 images belong to a person and the remaining 8 images are from another. The red rectangles show the wrong clustering results. From Figure 4, we observe that our method can well cluster the face images except the confusing ones.

Figure 5 presents the matrices \mathbf{Z}_0 , \mathbf{Z} and the visualizations of these two matrices with t-SNE [40] on the ORL dataset (we randomly selected 10 classes). The first row shows the matrices \mathbf{Z}_0 and \mathbf{Z} while the second row shows the clustering results with t-SNE by using \mathbf{Z}_0 and \mathbf{Z} . In Figure 5, different geometric figures belong to different categories and the clustering results are good when the same categories are close to each other. We observe that \mathbf{Z} has more discriminative power than \mathbf{Z}_0 because each category in \mathbf{Z} is more scattered from the visualizations. The specific representation matrices contain the individual information within each view and are beneficial to subspace clustering. The structural general representation matrix can preserve the intrinsic similarity of data which is also important to group data.

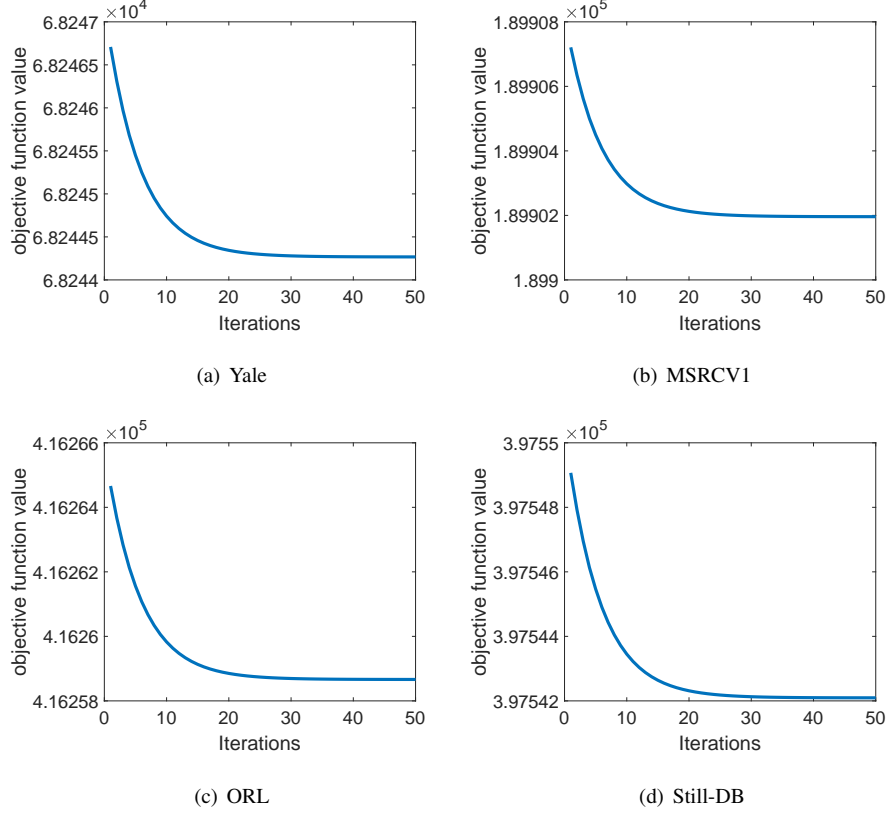


Figure 6: The convergence curves on benchmark datasets with $\lambda_1 = 0.001$, $\lambda_2 = 0.001$ and $\lambda_3 = 0.001$. The X-axis and Y-axis represent the number of iterations and the value of our objective function respectively.

5.2.5. Time Analysis

Table 7 shows the average computational time of different multi-view subspace clustering methods. From Tables 2-5, we observe that SMSC and ECMSC have good performance on benchmark datasets. However, ECMSC costs too much time. MVLRSS-C requires the least computational time on the Yale and ORL datasets, but the performance of MVLRSSC is poor compared with multi-view methods. SPC_{co-reg} has low time complexity on the MSRCV1 and Still-DB datasets. CSMSC solves the shared and individual matrices. However, no structural information is used to restrain the learnt matrices. Moreover, CSMSC has a high computational cost. DiMSC utilizes the diversity among each view and has few requirements for calculation. In general, our method

achieves a good trade-off between the performance and speed.

5.2.6. Convergence Analysis

400 We used the Augmented Lagrange Multiplier (ALM) method [31] to iteratively update all the pending matrices in our optimization problem and the convexity of the Lagrange function guarantees the effectiveness of our proposed method in some degree [14]. Figure 6 shows the objective function values on the Yale, MSRCV1, ORL and Still-DB four datasets. We observe that the values of our objective function on four
405 datasets decrease quickly in each iteration and approach to a point.

6. Conclusion

In this paper, we have presented a multi-view subspace clustering method. Our proposed method models the common and complementary information among each view via the general and specific representation matrices. The general representation
410 matrix holds the similarity relationship between the feature space and representation space while the specific representation matrices follow the diversity constraint. Due to this, our method is quite different from most existing multi-view subspace clustering methods which embed different views into a common space. Importantly, an effective algorithm is designed to handle the proposed optimization problem and the loss of
415 our objective function is proved to decrease. We conduct experiments on four popular datasets and give detailed analyses about the experimental results. Experimental results clearly demonstrate that our method can outperform the compared single and multiple view subspace clustering methods.

Our method can exploit the general and specific information between different
420 views. However, many issues exist to be addressed. Our method focuses on linear transformations among each view and nonlinear relationship can not be exploited. Otherwise, the prior information about the weight vector is not considered. For future work, we are interested in extending our method into a deep framework to exploit nonlinear information to further improve the performance of our method. Moreover, how
425 to discover the prior information about the quality of each view seems to be another interesting future direction.

Acknowledgement

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFA0700802, in part by the National Natural Science Foundation of China under Grant 61672306, Grant U1713214, Grant 61572271, and Grant 61527808, in part by the National 1000 Young Talents Plan Program, and in part by the Shenzhen Fundamental Research Fund (Subject Arrangement) under Grant JCYJ20170412170602564.

References

- [1] E. Elhamifar, R. Vidal, Sparse subspace clustering: Algorithm, theory, and applications, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35 (11) (2013) 2765–2781.
- [2] C. Lu, J. Feng, Z. Lin, T. Mei, S. Yan, Subspace clustering by block diagonal representation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41 (2) (2019) 487–501.
- [3] X. Wang, Z. Lei, X. Guo, C. Zhang, H. Shi, S. Z. Li, Multi-view subspace clustering with intactness-aware similarity, *Pattern Recognition* 88 (2019) 50–63.
- [4] W. Zhu, J. Lu, J. Zhou, Nonlinear subspace clustering for image clustering, *Pattern Recognition Letters* 107 (2018) 131–136.
- [5] Z. Ding, Y. Fu, Robust multi-view data analysis through collective low-rank subspace, *IEEE Transactions on Neural Networks and Learning Systems* 29 (5) (2018) 1986–1997.
- [6] J. Gu, L. Jiao, F. Liu, S. Yang, R. Wang, P. Chen, Y. Cui, J. Xie, Y. Zhang, Random subspace based ensemble sparse representation, *Pattern Recognition* 74 (2018) 544–555.
- [7] J. Wang, Z. Deng, K.-S. Choi, Y. Jiang, X. Luo, F.-L. Chung, S. Wang, Distance metric learning for soft subspace clustering in composite kernel space, *Pattern Recognition* 52 (2016) 113–134.

- [8] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, Y. Ma, Robust recovery of subspace structures by low-rank representation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35 (1) (2013) 171–184.
- [9] C.-G. Li, C. You, R. Vidal, Structured sparse subspace clustering: A joint affinity learning and subspace clustering framework, *IEEE Transactions on Image Processing* 26 (6) (2017) 2988–3001.
- [10] H. Gao, F. Nie, X. Li, H. Huang, Multi-view subspace clustering, in: *IEEE International Conference on Computer Vision*, 2015, pp. 4238–4246.
- [11] C. Tang, X. Zhu, X. Liu, M. Li, P. Wang, C. Zhang, L. Wang, Learning joint affinity graph for multi-view subspace clustering, *IEEE Transactions on Multimedia* (2018) 1–1.
- [12] M. Brbic, I. Kopriva, Multi-view low-rank sparse subspace clustering, *Pattern Recognition* 73 (2018) 247–258.
- [13] X. Cao, C. Zhang, H. Fu, S. Liu, H. Zhang, Diversity-induced multi-view subspace clustering, in: *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 586–594.
- [14] S. Luo, C. Zhang, W. Zhang, X. Cao, Consistent and specific multi-view subspace clustering, in: *AAAI Conference on Artificial Intelligence*, 2018, pp. 3730–3737.
- [15] S. T. Roweis, L. K. Saul, Nonlinear dimensionality reduction by locally linear embedding, *Science* 290 (5500) (2000) 2323–2326.
- [16] C. Zhang, H. Fu, Q. Hu, X. Cao, Y. Xie, D. Tao, D. Xu, Generalized latent multi-view subspace clustering, *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2018) 1–1.
- [17] C. Xu, D. Tao, C. Xu, Multi-view intact space learning, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37 (12) (2015) 2531–2544.
- [18] J. Zhao, X. Xie, X. Xu, S. Sun, Multi-view learning overview: recent progress and new challenges, *Information Fusion* 38 (2017) 43–54.

- [19] A. Kumar, H. Daumé, A co-training approach for multi-view spectral clustering, in: International Conference on International Conference on Machine Learning, 2011, pp. 393–400.
- [20] A. Kumar, P. Rai, H. Daume, Co-regularized multi-view spectral clustering, in: Advances in Neural Information Processing Systems, 2011, pp. 1413–1421.
- [21] R. Xia, Y. Pan, L. Du, J. Yin, Robust multi-view spectral clustering via low-rank and sparse decomposition., in: AAAI Conference on Artificial Intelligence, 2014, pp. 2149–2155.
- [22] M. Salzmänn, C. H. Ek, R. Urtasun, T. Darrell, Factorized orthogonal latent spaces, in: International Conference on Artificial Intelligence and Statistics, 2010, pp. 701–708.
- [23] Z. Ding, Y. Fu, Low-rank common subspace for multi-view learning, in: IEEE International Conference on Data Mining, 2014, pp. 110–119.
- [24] P. Zhu, W. Zhu, Q. Hu, C. Zhang, W. Zuo, Subspace clustering guided unsupervised feature selection, Pattern Recognition 66 (2017) 364–374.
- [25] Q. Li, Z. Sun, Z. Lin, R. He, T. Tan, Transformation invariant subspace clustering, Pattern Recognition 59 (2016) 142–155.
- [26] L. Chen, S. Wang, K. Wang, J. Zhu, Soft subspace clustering of categorical data with probabilistic distance, Pattern Recognition 51 (2016) 322–332.
- [27] V. R. De Sa, Spectral clustering with two views, in: International Conference on International Conference on Machine Learning workshop, 2005, pp. 20–27.
- [28] J. Ma, R. Wang, W. Ji, J. Zhao, M. Zong, A. Gilman, Robust multi-view continuous subspace clustering, Pattern Recognition Letters (2018) 1–1.
- [29] X. Peng, J. Feng, S. Xiao, W.-Y. Yau, J. T. Zhou, S. Yang, Structured autoencoders for subspace clustering, IEEE Transactions on Image Processing 27 (10) (2018) 5076–5086.

- [30] H. Hu, J. Feng, J. Zhou, Exploiting unsupervised and supervised constraints for subspace clustering, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37 (8) (2015) 1542–1557.
- 510 [31] Z. Lin, M. Chen, Y. Ma, The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices, *arXiv preprint arXiv:1009.5055*.
- [32] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al., Distributed optimization and statistical learning via the alternating direction method of multipliers, *Foundations and Trends® in Machine learning* 3 (1) (2011) 1–122.
- 515 [33] J.-F. Cai, E. J. Candès, Z. Shen, A singular value thresholding algorithm for matrix completion, *SIAM International Conference on Data Mining* 20 (4) (2010) 1956–1982.
- [34] R. Vidal, P. Favaro, Low rank subspace clustering (lsrc), *Pattern Recognition Letters* 43 (2014) 47–61.
- 520 [35] G. Liu, S. Yan, Active subspace: toward scalable low-rank learning, *Neural Computation* 24 (12) (2012) 3371–3394.
- [36] J. Yang, W. Yin, Y. Zhang, Y. Wang, A fast algorithm for edge-preserving variational multichannel image restoration, *SIAM International Conference on Data Mining* 2 (2) (2011) 569–592.
- 525 [37] Y. Liu, L. Jiao, F. Shang, An efficient matrix factorization based low-rank representation for subspace clustering, *Pattern Recognition* 46 (1) (2013) 284–292.
- [38] N. Ikizler, R. G. Cinbis, S. Pehlivan, P. Duygulu, Recognizing actions from still images, in: *International Conference on Pattern Recognition*, 2008, pp. 1–4.
- 530 [39] A. Y. Ng, M. I. Jordan, Y. Weiss, On spectral clustering: analysis and an algorithm, in: *Advances in Neural Information Processing Systems*, 2001, pp. 849–856.
- [40] L. v. d. Maaten, G. Hinton, Visualizing data using t-sne, *Journal of Machine Learning Research* 9 (2008) 2579–2605.