

Journal Pre-proof

Joint representation learning for multi-view subspace clustering

Guang-Yu Zhang, Yu-Ren Zhou, Chang-Dong Wang, Dong Huang,
Xiao-Yu He

PII: S0957-4174(20)30707-7
DOI: <https://doi.org/10.1016/j.eswa.2020.113913>
Reference: ESWA 113913

To appear in: *Expert Systems With Applications*

Received date : 12 April 2020
Revised date : 13 August 2020
Accepted date : 21 August 2020

Please cite this article as: G.-Y. Zhang, Y.-R. Zhou, C.-D. Wang et al., Joint representation learning for multi-view subspace clustering. *Expert Systems With Applications* (2020), doi: <https://doi.org/10.1016/j.eswa.2020.113913>.

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2020 Published by Elsevier Ltd.



Joint Representation Learning for Multi-view Subspace Clustering

Guang-Yu Zhang^a, Yu-Ren Zhou^{a,b,*}, Chang-Dong Wang^a,
Dong Huang^c, Xiao-Yu He^a

^a*School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China.*

^b*Engineering Research Institute, Guangzhou College of South China University of Technology, Guangzhou, China*

^c*College of Mathematics and Informatics, South China Agricultural University, Guangzhou, China.*

Abstract

Multi-view subspace clustering has made remarkable achievements in the field of multi-view learning for high-dimensional data. However, many existing multi-view subspace clustering methods still have two disadvantages. First, most of them only recover the subspace structure from either consistent or specific perspective. Second, they often fail to take advantage of the high-order information among different views. To alleviate these two issues, this paper proposes a novel multi-view subspace clustering method, which aims to learn the view-specific representation as well as the low-rank tensor representation in a unified framework. Particularly, our method learns the view-specific representation from data samples by exploiting the local structure within each view. In the meantime, we generate the low-rank tensor representation from the view-specific representation to capture the high-order correlation across multiple views. Based on the joint representation learning framework, the proposed method is able to explore the intra-view pairwise information and the inter-view complementary information, so that the underlying data structure can be revealed and then the final clustering result can be obtained through the subsequent spectral clustering. Furthermore, in the proposed Joint Representation Learning for Multi-view Subspace Clustering (JRL-MSC)

*Corresponding author.

Email addresses: guangyuzhg@foxmail.com (Guang-Yu Zhang),
zhouyuren@mail.sysu.edu.cn (Yu-Ren Zhou), changdongwang@hotmail.com
(Chang-Dong Wang), huangdonghere@gmail.com (Dong Huang),
hxyokokok@foxmail.com (Xiao-Yu He)

method, a unified objective function is formulated, which can be efficiently optimized by the alternating direction method of multipliers. Experimental results on multiple real-world data sets have demonstrated that our method outperforms the state-of-the-art counterparts.

Keywords: Multi-view subspace clustering, view-specific representation learning, low-rank tensor representation learning, unified framework

1. Introduction

With the development of information technology, multi-view data have become increasingly available in many scientific tasks. For instance, in computer vision, the image features may be extracted from various visual descriptors like LBP, SIFT and HOG. In natural language processing, each multilingual document may be reported in different languages, such as Chinese, English, French, and so on. In web information retrieval, a web page can be represented by two views, e.g., the visual images and the corresponding textual descriptions. As illustrated in these examples, multi-view data can be generated from diverse domains or views, where different views characterize the same object from heterogeneous viewpoints. In these situations, how to fuse the complementary knowledge from different views is still a challenging yet essential problem.

Over the past decade, a great number of multi-view clustering methods have been developed. Based on the different strategies, they can be roughly classified into four categories: k -means based methods Zhang et al. (2017b, 2018), matrix factorization based methods Zong et al. (2017); Nie et al. (2020), graph-based methods Li et al. (2015); Zhu et al. (2018) and subspace-based methods Brbić & Kopriva (2018); Huang et al. (2019c). Compared to the other methods, graph-based methods and subspace-based methods have attracted considerable attention owing to their superior performance and mathematical interpretability. The graph-based methods are originally derived from the traditional spectral learning models, which address the multi-view clustering problems by performing multiple graphs fusing and partitioning. Recent researchers have proposed several typical works in graph-based methods, such as Nie

et al. (2017a,b); Zhan et al. (2018); Liang et al. (2019). For these methods, the key is
 25 to learn a consistent similarity matrix with k connected components, so that the under-
 lying structure could be easily detected in the spectral space.

Beyond the graph-based methods, another alternative research focus for multi-view
 clustering resides in the subspace-based methods. In this category of methods, the pri-
 mary goal is to generate a meaningful subspace representation across multiple views
 30 with self-expression property. To achieve this goal, several subspace-based methods
 have been developed based on different strategies, including Cao et al. (2015); Zhang
 et al. (2017a); Wang et al. (2015). In the fields of data mining and computer vision,
 the aforementioned subspace-based methods have reported encouraging performance
 compared with other types of methods. However, they still suffer from one common
 35 drawback. That is, these methods neglect the high-order information across multi-
 ple views, which is essential to the multi-view subspace clustering. In more recent
 years, some researchers tend to solve this issue by introducing tensor factorization
 technology, like Zhang et al. (2015); Xie et al. (2018); Yin et al. (2018). Though these
 tensor subspace-based methods have made considerable progress, most of them tend
 40 to consider the global consistency among different views but ignore the specific local
 structure in each view.

To address the above issues, in this paper, we develop a new subspace-based method
 termed Joint Representation Learning for Multi-view Subspace Clustering (JRL-MSC).
 The key in the proposed method is to discover the intrinsic subspace structure through a
 45 unified representation learning scheme. Particularly, our method learns a set of smooth
 representations in the view-specific spaces, and simultaneously generates a tensor rep-
 resentation from the smooth representations by tensor-Singular Value Decomposition
 (t-SVD), which enables our method to make full use of the intra-view pairwise re-
 lationship and the inter-view high-order correlation Kilmer et al. (2013). The main
 50 contributions of this paper are summarized as follows:

- Our method learns the view-specific representation and the low-rank tensor rep-
 resentation in a unified framework. Both the specific information within each
 view and the complementary information among different views are captured

for enhancing the multi-view subspace clustering.

- 55 • We design an effective algorithm to solve the JRL-MSc optimization problem with computational complexity analysis.
- Comprehensive experiments are conducted on eight benchmark data sets, and the experimental results validate the effectiveness of our proposed joint representation learning scheme.

60 The rest of this paper is organized as follows. In Section 2, we briefly review some related works. In Section 3, we introduce some notations and preliminaries. And in Section 4, we introduce our methods, including the proposed model, related optimization algorithm and corresponding time complexity analysis. Comprehensive experimental results and discussions are provided in Section 5. Finally, we provide the
65 conclusion and future work in Section 6.

2. Related Work

Multi-view clustering is a very powerful tool in analyzing data with heterogeneous features. During the last two decades, numerous multi-view clustering methods have been proposed to achieve robust clustering performance. In what follows, we will
70 briefly introduce several multi-view clustering methods from different perspectives.

K -means based methods are one of the earliest schemes for multi-view clustering. To name a few, Tzortzis & Likas (2012) developed a weighted kernel k -means method for multi-view clustering, which achieved success in various nonlinearly separable data sets. After that, Chen et al. (2013) developed a classical multi-view k -means clustering
75 method termed TW- k -means, which is able to compute the weights of different views and individual features simultaneously. Besides, Cai et al. (2013) designed a novel k -means clustering method for multi-view big data, whose computational speed is similar to the k -means clustering algorithm. In addition, Jiang et al. (2014) extended collaborative fuzzy k -means to multi-view domain and proposed a weighted clustering method
80 named WV-Co-FCM. Xu et al. (2015) further presented an interesting k -means based method to deal with the high-dimensional cases. In this method, the objective is to

integrate multi-view clustering and feature selection into a joint framework, so that the suitable views and important features could be selected for clustering.

Matrix factorization based methods belong to another category of multi-view clustering, and they have a strong connection with k -means methods. Liu et al. (2013b) applied nonnegative matrix factorization to multi-view clustering and developed a well-known method termed MultiNMF. In this method, the key idea is to discover the latent common factors across different views by joint matrix factorization process. Thereafter, Zhao et al. (2017) further presented a deep multi-view clustering method based on semi-nonnegative factorization, where a deep structure is built to capture the common representation from multiple views. Following this work, Huang et al. (2019c) developed a multi-view clustering method to uncover the hierarchical semantics across different views. Moreover, Huang et al. (2019b) made an effort to solve the view-insufficiency issue and designed a robust clustering method called MVIC. Recently, Yao et al. (2019) proposed a novel method based on multiple co-clustering strategy, which aims to generate a series of high-quality clusterings by considering both individuality and commonality of multi-view data.

Graph-based methods play a central role in the context of multi-view clustering, which provide an effective way to solve the nonlinearly separated problems. For example, Nie et al. (2017a) developed a popular graph-based method, which is capable to perform multi-view clustering and local structure learning simultaneously. At the same time, Nie et al. (2017b) proposed another graph-based method for multi-view clustering named SwMC. The basic idea behind this method is to generate a robust common graph from a series of low-quality view-specific graphs. Similarly, Zhan et al. (2017) further designed a notable clustering method based on two-step multiple graph fusion strategy. What's more, Li et al. (2018) proposed a novel method that performs multi-view graph partitioning with discriminative metric learning. In this method, both the intra-view connections and the inter-view couplings are taken into consideration to facilitate the clustering performance.

Subspace-based methods have recently become the mainstream in multi-view clustering researches, aiming at discovering the latent subspace structure across different views. Gao et al. (2015) made an early attempt to extend traditional subspace-based

method for multi-view clustering, termed as MVSC. Unlike the previous methods, MVSC not only captures the subspace structure in each view, but also guarantees the consistency among different views. Along this line, Wang et al. (2017) further developed a subspace-based method that exploits both representation complementarity as well as indicator consistency for multi-view clustering. Moreover, Zhang et al. (2017a) proposed a novel multi-view subspace clustering method, whose key idea is to conduct subspace clustering in the latent common space. In addition, Zhang et al. (2020) designed another multi-view subspace clustering method based on one-step framework. Very recently, Chen et al. (2020) developed a similar subspace clustering method named MCLES. To boost the clustering performance, this method jointly constructs the latent embedding space from multiple views, while at the same time discovers the latent subspace structure in the learned embedding space. Apart from the aforementioned methods, several efforts have been made on the tensor subspace-based methods in recent years. For example, Zhang et al. (2015) presented a multi-view subspace clustering method based on unfolding low-rank tensor constraint. To further enhance the physical meaning, Xie et al. (2018) introduced a new low-rank tensor constraint on the rotated tensor, and thereby established a novel method named t-SVD-MSC. More recently, Yin et al. (2018) developed a novel multi-view subspace clustering method by means of t-product operation. Particularly, this method organizes the multi-view features as a third-order tensor and simultaneously exploits the tensor subspace structure through sparse and low-rank tensor constraints.

3. Notations and Preliminaries

This section briefly introduces some notations and preliminaries, which are related to the proposed method.

Throughout this paper, we denote tensors by calligraphy letters (e.g., $\mathcal{A}$), matrices by capital letters (e.g. A), and vectors by lowercase letters (e.g. a). For a third-order tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, its i -th horizontal slice, j -th lateral slice and k -th frontal slice are defined by $\mathcal{A}(i, :, :)$, $\mathcal{A}(:, j, :)$ and $\mathcal{A}(:, :, k)$, respectively. The Discrete Fourier transform (DFT) along the third dimension of tensor $\mathcal{A}$ is denoted-

ed by $\mathcal{A}_f = \text{fft}(\mathcal{A}, [], 3)$. Conversely, tensor $\mathcal{A}$ can be obtained by inverse DFT, i.e., $\mathcal{A} = \text{ifft}(\mathcal{A}_f, [], 3)$. Moreover, we use matrix $A^{(k)}$ to represent the frontal slice $\mathcal{A}(:, :, k)$ for brevity.

145 Before introducing the definition of t-SVD, we need to give the five block-based operators, namely, bcirc, unfold, fold, bdiag and bdfold, in advance. For any third-order tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, the block circulant matrix $\text{bcirc}(\mathcal{A}) \in \mathbb{R}^{n_1 n_3 \times n_1 n_3}$ is defined as follows.

$$\text{bcirc}(\mathcal{A}) = \begin{bmatrix} A^{(1)} & A^{(n_3)} & \dots & A^{(2)} \\ A^{(2)} & A^{(1)} & \dots & A^{(3)} \\ \vdots & \ddots & \ddots & \vdots \\ A^{(n_3)} & A^{(n_3-1)} & \dots & A^{(1)} \end{bmatrix}.$$

And the unfold as well as fold operations of tensor $\mathcal{A}$ can be defined as follows.

$$\text{unfold}(\mathcal{A}) = \begin{bmatrix} A^{(1)} \\ A^{(2)} \\ \vdots \\ A^{(n_3)} \end{bmatrix}, \text{fold}(\text{unfold}(\mathcal{A})) = \mathcal{A}.$$

150 Besides, the bdiag and bdfold operations of tensor $\mathcal{A}$ are defined as follows.

$$\text{bdiag}(\mathcal{A}) = \begin{bmatrix} A^{(1)} & & & \\ & A^{(2)} & & \\ & & \ddots & \\ & & & A^{(n_3)} \end{bmatrix}, \text{bdfold}(\text{bdiag}(\mathcal{A})) = \mathcal{A}.$$

Next, we give some important definitions that are related to t-SVD in what follows.

Definition 1. (t-product Kilmer & Martin (2011)). Let $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, and $\mathcal{B} \in \mathbb{R}^{n_2 \times n_4 \times n_3}$. Then the t-product $\mathcal{C} = \mathcal{A} * \mathcal{B}$ is a tensor of size $n_1 \times n_4 \times n_3$,

$$\mathcal{C} = \mathcal{A} * \mathcal{B} = \text{fold}(\text{bcirc}(\mathcal{A}) \cdot \text{unfold}(\mathcal{B})). \quad (1)$$

Note that the t-product represents the multiplication between two tensors, which is analogous to the matrix multiplication except that the circular convolution replaces the multiplication operation between the elements, i.e.,

$$\mathcal{C}(i, j, :) = \sum_{k=1}^{n_2} \mathcal{A}(i, k, :) \circ \mathcal{B}(k, j, :). \quad (2)$$

Here " $\circ$ " represents the circular convolution between two tubes. To be specific, t-product in the original domain is equivalent to the matrix multiplication in the Fourier domain, which can be further expressed as follows:

$$\mathcal{C}_f^{(k)} = \mathcal{A}_f^{(k)} \mathcal{B}_f^{(k)}, k = 1, \dots, n_3, \quad (3)$$

where $\mathcal{C}_f$, $\mathcal{A}_f$ and $\mathcal{B}_f$ are the Discrete Fourier Transformation (DFT) of tensors $\mathcal{C}$, $\mathcal{A}$ and $\mathcal{B}$, respectively.

Definition 2. (f-diagonal tensor Kilmer & Martin (2011)). A tensor is called f-diagonal if each of its frontal slices is diagonal matrix.

Definition 3. (Identity tensor Kilmer & Martin (2011)). For the identity tensor $\mathcal{I} \in \mathbb{R}^{n \times n \times n_3}$, its first frontal slice is the identity matrix with size $n \times n$, and all other frontal slices are zero.

Definition 4. (Tensor transpose Kilmer & Martin (2011)). For any tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, its transpose tensor $\mathcal{X}^T \in \mathbb{R}^{n_2 \times n_1 \times n_3}$ can be obtained by transposing each frontal slice of $\mathcal{X}$ and then reversing the order of the transposed frontal slices 2 through n_3 .

Definition 5. (Orthogonal tensor Kilmer & Martin (2011)). A tensor $\mathcal{Q} \in \mathbb{R}^{n \times n \times n_3}$ is orthogonal if it satisfies

$$\mathcal{Q}^T * \mathcal{Q} = \mathcal{Q} * \mathcal{Q}^T = \mathcal{I}, \quad (4)$$

where " $*$ " denotes the t-product.

Theorem 1. (t-SVD Kilmer & Martin (2011)). For a tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, it can be factorized by t-SVD as

$$\mathcal{A} = \mathcal{U} * \mathcal{S} * \mathcal{V}^T, \quad (5)$$

where $\mathcal{U} \in \mathbb{R}^{n_1 \times n_1 \times n_3}$ and $\mathcal{V} \in \mathbb{R}^{n_2 \times n_2 \times n_3}$ are orthogonal, and $\mathcal{S} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is
 165 f -diagonal. The specific proof process of t -SVD can be found in paper Kilmer & Martin (2011).

Especially, Figure 1 shows the t -SVD sample of a tensor $\mathcal{X}$. Based on the t -operation demonstrated in Eq. (3), we can compute the t -SVD of a given tensor $\mathcal{A}$ in the Fourier domain, which is shown in Algorithm 1 Kilmer et al. (2013).

Algorithm 1 t -SVD

Input: Tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$.

- 1: $\mathcal{A}_f = \text{fft}(\mathcal{A}, [], 3)$.
- 2: **for** $k = 1: n_3$ **do**
- 3: $[\mathbf{U}, \Sigma, \mathbf{V}] = \text{SVD}(\mathcal{A}_f^{(k)})$.
- 4: $\mathcal{U}_f^{(k)} = \mathbf{U}$, $\mathcal{S}_f^{(k)} = \Sigma$, $\mathcal{V}_f^{(k)} = \mathbf{V}$.
- 5: **end**
- 6: $\mathcal{U} = \text{ifft}(\mathcal{U}_f, [], 3)$, $\mathcal{S} = \text{ifft}(\mathcal{S}_f, [], 3)$, $\mathcal{V} = \text{ifft}(\mathcal{V}_f, [], 3)$.

Output: Tensors $\mathcal{U}$, $\mathcal{S}$ and $\mathcal{V}$.

Definition 6. (t -SVD based tensor nuclear norm). The t -SVD based tensor nuclear norm $\|\mathcal{A}\|_{\oplus}$ of a tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is defined by the sum of singular values of all the frontal slices of $\mathcal{A}_f$:

$$\|\mathcal{A}\|_{\oplus} = \sum_{k=1}^{n_3} \|\mathcal{A}_f^{(k)}\|_* = \sum_{i=1}^{\min(n_1, n_2)} \sum_{k=1}^{n_3} |S_f^{(k)}(i, i)|, \quad (6)$$

170 where $\mathcal{A}_f$ is the Discrete Fourier Transformation (DFT) of tensor $\mathcal{A}$. $S_f^{(k)}(i, i)$ can be computed by the t -SVD as $\mathcal{A}_f^{(k)} = \mathcal{U}_f^{(k)} \mathcal{S}_f^{(k)} \mathcal{V}_f^{(k)T}$ of frontal slices of $\mathcal{A}_f$ in the Fourier domain.

4. Methodology

4.1. Background

175 This subsection briefly reviews the general framework for conventional subspace clustering, including a representative method, namely, Subspace Clustering by SMOoth Representation (SMR) Hu et al. (2014).

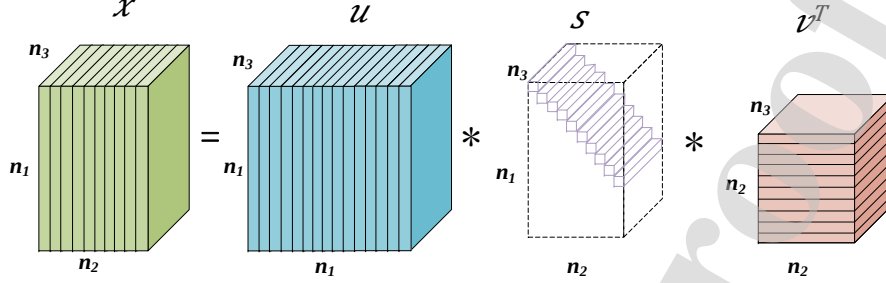


Figure 1: The t-SVD sample of a tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$.

In the present literature, the mainstream for subspace clustering falling into the self-expressive based methods. However, beyond this category of methods, several other subspace clustering approaches have been developed over the past years, such as Greedy Subspace Clustering Park et al. (2014), Robust Subspace Clustering via Thresholding Heckel & Bölcskei (2015), Innovation Pursuit based Subspace Clustering Rahmani & Atia (2017), Subspace Clustering using Ensembles of K-Subspaces (EKSS) Lipor et al. (2017) and Adaptive online k-subspaces with cooperative re-initialization Lane et al. (2019). The aforementioned subspace clustering methods address the clustering problems based on different strategies, which have shown encouraging performance in many pattern recognition and computer vision tasks, due to their strong theoretical and empirical guarantees.

In this paper, we focus on the self-expressive based methods for multi-view subspace clustering. For the sake of convenience, we will abbreviate the self-expressive based subspace clustering as subspace clustering in the rest of our paper. Formally, let $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\} \in \mathbb{R}^{d \times n}$ be a given data set, where each column represents a data sample, and d denotes the dimensionality of the feature space. The purpose of subspace clustering methods is to partition the data samples into several linear subspaces by using self-expression property. Particularly, self-expression property indicates that the data samples can be expressed by the linear combinations of themselves. Without

loss of generality, we can write this in the matrix form:

$$X = XZ + E, \quad (7)$$

where $Z \in \mathbb{R}^{n \times n}$ is the representation matrix and $E \in \mathbb{R}^{d \times n}$ is the error matrix. In the light of this, the general optimization problem of subspace clustering can be written as follows:

$$\begin{aligned} \min_{Z, E} \quad & \Theta(X, XZ) + \alpha \Omega(Z) \\ \text{s.t.} \quad & X = XZ + E. \end{aligned} \quad (8)$$

Here $\Theta(\cdot, \cdot)$ and $\Omega(\cdot)$ represent the loss function and regularized term, respectively.

$\alpha > 0$ represents a balanced parameter that is controlled by users. After obtaining the representation matrix Z , we can compute the affinity matrix by $A = (|Z| + |Z^T|)/2$, and further employ it as the input of the spectral clustering algorithm for achieving the final clustering result.

In the last decade, researchers have designed several representative subspace clustering methods from different perspectives, such as Sparse Subspace Clustering (SSC) Elhamifar & Vidal (2013), Low-Rank Recovery (LRR) Liu et al. (2012), Subspace Clustering by SMOOTH Representation (SMR) Hu et al. (2014) and Low-Rank Sparse Subspace Clustering (LRSSC) Wang et al. (2013). Among these methods, Subspace Clustering by SMOOTH Representation (SMR) has become one of the most popular methods owing to its promising performance and scalability. This method not only explores the global subspace structure, but also preserves the local geometrical structure for better grouping the data samples into correct subspace. For that goal, it needs to learn a smooth representation by enhancing the group effect of representations. Accordingly, the regularized term for SMR is defined as follows:

$$\text{Tr}(ZLZ^T) = \sum_{i=1}^n \sum_{j=1}^n \|z_i - z_j\|_2^2 w_{ij}, \quad (9)$$

where $W \in \mathbb{R}^{n \times n}$ is the graph weighting symmetric matrix that encodes the local similarity information among data samples. Besides, $L = D - W$ is the Laplacian matrix, where D is the diagonal matrix and its element $d_{ii} = \sum_{j=1}^n w_{ij}$. Consider

an input data matrix $X = \{x_1, \dots, x_n\}$, the target function of SMR is formulated as follows:

$$\begin{aligned} \min_{Z, E} \quad & \|E\|_F^2 + \alpha \text{Tr}(ZLZ^T) \\ \text{s.t.} \quad & X = XZ + E. \end{aligned} \quad (10)$$

In practice, there are many ways to construct the graph weighting matrix, like 0-1 weighting, heat kernel weighting and so on. Different from the SMR method, we choose the heat kernel weighting strategy to construct the graph weighting matrices due to its effectiveness.

4.2. The proposed model

This subsection presents the details of our proposed model step by step. In the multi-view setting, suppose that we are given a data set $X = [X^{(1)^T}, \dots, X^{(V)^T}]^T$ with V views, where $X^{(v)} = [x_1^{(v)}, \dots, x_n^{(v)}] \in \mathbb{R}^{d_v \times n}$ denotes the v -th data matrix with d_v dimensionality. Intuitively, self-expression property can be easily extended to the multi-view scenario in a direct way. If we treat each view independently, the multi-view self-expression formulation could be expressed as follows:

$$X^{(v)} = X^{(v)}Z^{(v)} + E^{(v)}, \forall v, \quad (11)$$

where $Z^{(v)} \in \mathbb{R}^{n \times n}$ is the representation matrix in the v -th view, and $E^{(v)} \in \mathbb{R}^{d_v \times n}$ is the error matrix in the v -th view. Based on this, we can further formulate the objective function of SMR based Multi-view Subspace Clustering (SMR-MS) as follows:

$$\begin{aligned} \min_{E^{(v)}, Z^{(v)}} \quad & \sum_{v=1}^V \|E^{(v)}\|_{2,1} + \lambda_1 \sum_{v=1}^V \text{Tr}(Z^{(v)}L^{(v)}Z^{(v)T}) \\ \text{s.t.} \quad & X^{(v)} = X^{(v)}Z^{(v)} + E^{(v)}, v = 1, \dots, n, \end{aligned} \quad (12)$$

where $L^{(v)}$ is the Laplacian matrix constructed in the v -th view, and λ_1 is a positive balancing parameter. Note that the above optimization problem (12) performs SMR in each view independently, which can not explore the complementarity and high-order information across different views. Hence, it is necessary to find an effective way to explore the high-order correction across different views. Many recent works Xie et al.

(2018); Yin et al. (2018); Wu et al. (2019, 2020) have validated the t-SVD based technology is able to sufficiently capture the low-rank structure across different views in the tensor space. Furthermore, t-SVD technology can learn the robust low-rank tensor representation which provides deeper insight into the multi-view features, comparing with the other tensor decomposition technologies. Motivated by the aforementioned benefits of t-SVD technology, we can further achieve the low-rank tensor representation $\mathcal{S}$ from original view-specific representation $\{Z^{(v)}\}_{v=1}^V$ by:

$$\begin{aligned} \min_{\mathcal{S}} & \|\mathcal{S}\|_{\otimes} + \frac{\lambda_3}{2} \|\mathcal{S} - \mathcal{Z}\|_F^2 \\ \text{s.t. } & \mathcal{Z} = \Phi(Z^{(1)}, Z^{(2)}, \dots, Z^{(V)}). \end{aligned} \quad (13)$$

Here $\Phi(\cdot)$ is an operator which stacks the view-specific representation $\{Z^{(v)}\}_{v=1}^V$ into a third-order tensor $\mathcal{Z}$, and then rotates its size to $n \times V \times n$, as shown in Figure 2. $\|\mathcal{S}\|_{\otimes}$ represents the t-SVD based tensor norm on tensor $\mathcal{S}$, as shown in Definition 7. To be specific, this introduced tensor norm ensures that our method can efficiently explore the high-order correlations of different views. Naturally, by putting the functions in Eq. (12) and Eq. (13) together, we eventually obtain the overall model of our method as follows:

$$\begin{aligned} \min_{Z^{(v)}, E^{(v)}, \mathcal{S}} & \sum_{v=1}^V \|E^{(v)}\|_{2,1} + \lambda_1 \sum_{v=1}^V \text{Tr}(Z^{(v)} L^{(v)} Z^{(v)T}) + \lambda_2 \|\mathcal{S}\|_{\otimes} + \frac{\lambda_3}{2} \|\mathcal{S} - \mathcal{Z}\|_F^2 \\ \text{s.t. } & X^{(v)} = X^{(v)} Z^{(v)} + E^{(v)}, \forall v, \\ & \mathcal{Z} = \Phi(Z^{(1)}, Z^{(2)}, \dots, Z^{(V)}), \end{aligned} \quad (14)$$

where λ_1 , λ_2 and λ_3 are three positive balancing parameters. Different from the existing methods, our method aims to learn the view-specific representation as well as the low-rank tensor representation in a joint framework. Under this joint representation learning framework, both the intra-view pairwise relationship and the inter-view correlation are fully explored for multi-view subspace clustering, and thus a more robust description of latent clustering structure should be provided. Particularly, in our method, the t-SVD tensor factorization is a very powerful tool to find the joint representation, so that the optimal affinity matrix could be obtained for generating the final

clustering results.

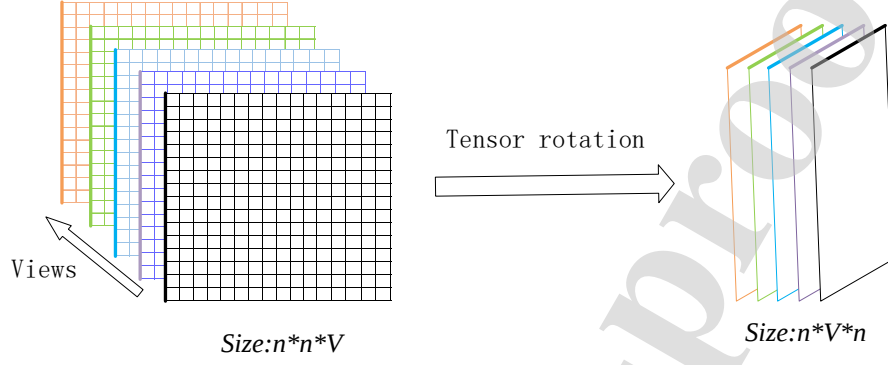


Figure 2: Example of rotation operation.

215 4.3. The optimization algorithm

The proposed model in Eq. (14) can be efficiently optimized by Alternating Direction Method of Multiplier (ADMM) algorithm Boyd et al. (2011). First of all, we need to convert it into the following augmented Lagrangian function:

$$\begin{aligned}
 & \mathcal{L}_{\mu>0}(\{Z^{(v)}\}_{v=1}^V, \{E^{(v)}\}_{v=1}^V, \mathcal{S}) \\
 &= \sum_{v=1}^V \|E^{(v)}\|_{2,1} + \lambda_1 \sum_{v=1}^V \text{Tr}(Z^{(v)} L^v Z^{(v)T}) + \lambda_2 \|\mathcal{S}\|_{\oplus} + \frac{\lambda_3}{2} \|\mathcal{S} - \mathcal{Z}\|_F^2 + \\
 & \sum_{v=1}^V \left\langle Y^{(v)}, X^{(v)} - X^{(v)} Z^{(v)} - E^{(v)} \right\rangle + \frac{\mu}{2} \sum_{v=1}^V \|X^{(v)} - X^{(v)} Z^{(v)} - E^{(v)}\|_F^2,
 \end{aligned} \tag{15}$$

where $Y^{(v)}$ is the Lagrange multiplier in the v -th view, and μ is the penalty parameter.

220 $\langle \cdot, \cdot \rangle$ means the inner product of any two matrices. Under the ADMM framework, we present an alternative optimization algorithm to solve this problem. Precisely, we can update one variable by treating the others as constant in one iteration. The corresponding description of optimization procedure is introduced in what follows.

4.3.1. Update rule for the variable $\{Z^{(v)}\}_{v=1}^V$

When the other variables $\{E^{(v)}\}_{v=1}^V$ and $\mathcal{S}$ are fixed, the optimization problem in Eq. (15) reduces to (Note that we have $\Phi_{(v)}^{-1}(\mathcal{S}) = S^{(v)}$ and $\Phi_{(v)}^{-1}(\mathcal{Z}) = Z^{(v)}$ here) :

$$\begin{aligned} \min_{Z^{(v)}} & \lambda_1 \text{Tr}(Z^{(v)} L^v Z^{(v)T}) + \frac{\lambda_3}{2} \|S^{(v)} - Z^{(v)}\|_F^2 + \\ & \langle Y^{(v)}, X^{(v)} - X^{(v)} Z^{(v)} - E^{(v)} \rangle + \frac{\mu}{2} \|X^{(v)} - X^{(v)} Z^{(v)} - E^{(v)}\|_F^2. \end{aligned} \quad (16)$$

The problem in Eq. (16) is a smooth convex program. Thus, we set the derivative of this problem w.r.t. variable $Z^{(v)}$ to zero and have:

$$(\lambda_3 I + \mu X^{(v)T} X^{(v)}) Z^{(v)} + Z^{(v)} (2\lambda_1 L^{(v)}) = \lambda_3 S^{(v)} + Q^{(v)}, \quad (17)$$

in which $Q^{(v)} = X^{(v)T} Y^{(v)} + \mu X^{(v)T} X^{(v)} - \mu X^{(v)T} E^{(v)}$. Apparently, the above equation is a Sylvester equation Bartels & Stewart (1972), and it can be efficiently solved by classical Bartels-Stewart algorithm Bartels & Stewart (1972). The core idea of Bartels-Stewart algorithm relies on applying the Schur decomposition to transform the Sylvester matrix equation into a triangular system, so that it can be efficiently solved by the backward substitutions on blocks. In our method, we use the `lyap` function of the **Matlab Control Toolbox** to solve the problem in Eq.(17).

4.3.2. Update rule for the variable $\{E^{(v)}\}_{v=1}^V$

When the other variables $\{Z^{(v)}\}_{v=1}^V$ and $\mathcal{S}$ are fixed, the optimization problem in Eq. (15) reduces to:

$$\min_{E^{(v)}} \|E^{(v)}\|_{2,1} + \frac{\mu}{2} \|E^{(v)} - (X^{(v)} - X^{(v)} Z^{(v)} + \frac{1}{\mu} Y^{(v)})\|_F^2. \quad (18)$$

The problem in Eq.(18) can be solved by Lemma 4.1 in (Liu et al. (2013a)). Concretely, the Lemma is introduced as follows:

Lemma 1. (Liu et al. (2013a)) For any given matrix F and parameters $\alpha, \beta > 0$, the optimal solution to the following problem

$$\min_D \alpha \|D\|_{2,1} + \frac{\beta}{2} \|D - F\|_F^2 \quad (19)$$

is given by

$$D_{:,j} = \begin{cases} \frac{\|F_{:,j}\|_2 - \frac{\alpha}{\beta}}{\|F_{:,j}\|_2} F_{:,j}, & \text{if } \|F_{:,j}\|_2 > \frac{\alpha}{\beta} \\ 0, & \text{otherwise.} \end{cases} \quad (20)$$

235 Here $D_{:,j}$ and $F_{:,j}$ represent the j -th column of matrix D and F , respectively. The proof of this lemma can be found in paper Yang et al. (2009). Based on this, the optimal solution to the problem in Eq.(18) is given by:

$$E_{:,j}^{(v)} = \begin{cases} \frac{\|H_{:,j}^{(v)}\|_2 - \frac{1}{\mu}}{\|H_{:,j}^{(v)}\|_2} H_{:,j}^{(v)} & \text{if } \|H_{:,j}^{(v)}\|_2 > \frac{1}{\mu}, \\ 0 & \text{otherwise.} \end{cases} \quad (21)$$

Here $H^{(v)} = X^{(v)} - X^{(v)}Z^{(v)} + \frac{1}{\mu}Y^{(v)}$ and $H_{:,j}^{(v)}$ denotes the j -th column of matrix $H^{(v)}$.

240 4.3.3. Update rule for the variable $\mathcal{S}$

When the other variables $\{Z^{(v)}\}_{v=1}^V$ and $\{E^{(v)}\}_{v=1}^V$ are fixed, the optimization problem in Eq. (15) reduces to:

$$\min_{\mathcal{S}} \lambda_2 \|\mathcal{S}\|_{\otimes} + \frac{\lambda_3}{2} \|\mathcal{S} - \mathcal{Z}\|_F^2. \quad (22)$$

The problem in Eq. (22) is referred to as the tensor multi-rank minimization problem. According to the Theorem 2 in Xie et al. (2018), we can obtain the solution of this problem by using the tensor tubal shrinkage operator:

$$\mathcal{S} = \mathcal{C}_{n_3\pi}(\mathcal{Z}) = \mathcal{U} * \mathcal{C}_{n_3\pi}(\mathcal{O}) * \mathcal{V}^T, \quad (23)$$

where $\pi = \lambda_3/\lambda_2$ and $\mathcal{Z} = \mathcal{U} * \mathcal{O} * \mathcal{V}^T$. Besides, $\mathcal{C}_{n_3}(\mathcal{Z}) = \mathcal{Z} * J$, herein, J is an $n_1 \times n_2 \times n_3$ f-diagonal tensor, and its diagonal element in the Fourier domain is $\mathcal{J}_f(i, i, j) = (1 - \frac{n_3\pi}{\mathcal{O}_f^{(j)}(i; i)})_+$. Note that the tubal shrinkage operator in the original domain is equivalent to the tensor SVT in the Fourier domain. Therefore, the problem

245 in Eq. (22) can be solved by an efficient algorithm 2.

Algorithm 2 t-SVD based Tensor Multi-Rank Minimization Xie et al. (2018)

Input: Tensor $\mathcal{Z} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, parameter τ (In problem (22), $\tau = \lambda_3/\lambda_2$).

Output: Tensor $\mathcal{S}$.

- 1: $\mathcal{Z}_f = \text{fft}(\mathcal{Z}, [], 3), \tilde{\tau} = n_3\tau$.
 - 2: **for** $k = 1: n_3$ **do**
 - 3: $[\mathcal{U}_f^{(k)}, \mathcal{C}_f^{(k)}, \mathcal{V}_f^{(k)}] = \text{SVD}(\mathcal{Z}_f^{(k)})$.
 - 4: $\mathcal{J}_f^{(k)} = \text{diag}\{(1 - \frac{\tilde{\tau}}{\mathcal{C}_f^{(k)}(i,i)})_+, i = 1, \dots, \min(n_1, n_2)\}$.
 - 5: $\mathcal{C}_{f,\tilde{\tau}}^{(k)} = \mathcal{C}_f^{(k)} \mathcal{J}_f^{(k)}$.
 - 6: $\mathcal{S}_f^{(k)} = \mathcal{U}_f^{(k)} \mathcal{C}_{f,\tilde{\tau}}^{(k)} \mathcal{V}_f^{(k)T}$.
 - 7: **end**
 - 8: $\mathcal{S} = \text{ifft}(\mathcal{S}_f, [], 3)$.
 - 9: **Return:** Tensors $\mathcal{S}$.
-

4.3.4. Update rule for the Lagrange multiplier $\{\mathbf{Y}^{(v)}\}_{v=1}^V$

In addition, we update the Lagrange multiplier by:

$$\mathbf{Y}^{(v)} = \mathbf{Y}^{(v)} + \mu(\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}^{(v)} - \mathbf{E}^{(v)}). \quad (24)$$

The above four steps are repeated until the stopping criterion is met or the number of iterations reaches the threshold value. Specifically, the stopping criterion is given as follows:

$$\sum_{v=1}^V \|\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}^{(v)} - \mathbf{E}^{(v)}\|_{\infty} < \varepsilon. \quad (25)$$

Here ε is a very small predefined value (i.e., 10^{-6} in this paper). Note that when the convergence condition in Eq. (25) is satisfied, we construct the final affinity matrix from the tensor representation $\mathcal{S}$ by unfolding it into a set of view-specific matrices (i.e., $\{\mathbf{S}^{(v)}\}_{v=1}^V$). For clarity, we summarize the whole optimization procedure in Algorithm 3.

4.4. Complexity analysis

As shown in Algorithm 3, the optimization procedure for our model mainly consists of four steps. In the first step, the computational complexity of updating variable $\mathbf{Z}^{(v)}$

Algorithm 3 JRL-MSC

Input: Multi-view data set $X = \{X^{(1)}, \dots, X^{(V)}\}$ with V views, the maximum number of iterations t_{max} (i.e., 10 here) and three balancing parameters $\lambda_1, \lambda_2, \lambda_3 > 0$.

Parameter Setup: Set the number of k -nearest neighbor $k_{nn} = 5$. Set parameters $\mu = 1$, $\rho = 1.5$ and $\mu_{max} = 10^6$.

- 1: **Initialization:** Set $t = 1$. Initialize $Z = \mathbf{0}$, $S = \mathbf{0}$ and $Y^{(v)} = \mathbf{0}, \forall v$.
- 2: **repeat**
- 3: Update $Z^{(v)}$ in the v -th view by using Bartels-Stewart algorithm to solve Sylvester equation (17).
- 4: Update $E^{(v)}$ in the v -th view by using Eq. (21).
- 5: Update $\mathcal{S}$ by using Algorithm 2.
- 6: Update $Y^{(v)}$ in the t -th view by using Eq. (24).
- 7: Update μ by using $\mu = \min(\rho\mu, \mu_{max})$.
- 8: $t = t + 1$.
- 9: **until** The stopping criterion in (25) is met or $t > t_{max}$
- 10: Obtain the overall affinity matrix by using: $A = \frac{1}{V} \sum_{v=1}^V \frac{|S^{(v)}| + |S^{(v)}|^T}{2}$.
- 11: Perform spectral clustering on the overall affinity matrix A and generate the final clustering label.

Output: The final clustering label.

255 is $O(n^3)$, where n represents the number of data samples. In the second step, the computational complexity of updating variable $E^{(v)}$ is $O(d_v n)$, where d_v represents the dimensionality in the t -th view. In the third step, the computational complexity of updating variable Z is $O(n \log n)$. In the fourth step, the computational complexity of updating multiplier $Y^{(v)}$ is $O(1)$. Therefore, the whole computational complexity of our method is about $O(T \times (\sum_{v=1}^V (n^3 + d_v n + 1) + n \log n)) = O(T \times n \times \max\{Vn^2, \sum_{v=1}^V d_v\})$, herein, T represents the iteration number and V represents the view number.

260

5. Experiments

In this section, we carry out extensive experiments to evaluate some properties of our proposed method: comparison experiments, parameter analysis and running time analysis. All the experiments are implemented on a workstation with Intel 3.4-GHz processors and 64-GB RAM.

5.1. Data Sets and Evaluation Metrics

The experiments are conducted on eight publicly available data sets: UCI Digit, MSRC-v1, Yale, YaleB, ORL, VIS/NIR, Reuters and Cora. They are collected from different applications: handwritten digits, object images, facial images, news reports and scientific publications. Table 1 describes the characterizes of these data sets. The information of these data sets is briefly introduced as follows.

Table 1: The statistics of eight data sets used in the experiments.

View type	UCI Digit	MSRC-v1	ORL	Yale	YaleB	VIS/NIR	Reuters	Cora
1	FOU(76)	CM(24)	Intensity(4096)	Intensity(4096)	Intensity(2500)	View 1(10000)	English(9749)	Cont(3000)
2	FAC(216)	GIST(512)	LBP(3304)	LBP(3304)	LBP(3304)	View 2(10000)	French(9109)	Cite(1840)
3	KAR(64)	LBP(256)	Gabor(6750)	Gabor(6750)	Gabor(6750)	-	German(7774)	-
4	PIX(240)	GENT(254)	-	-	-	-	-	-
5	ZER(47)	-	-	-	-	-	-	-
6	SIFT(200)	-	-	-	-	-	-	-
Samples	2000	210	400	165	650	1056	600	1051
Classes	10	7	40	15	10	22	6	2

- **UCI Digit**¹ is a multi-view data set for handwritten digits (0-9) recognition. In this data set, there are 2000 samples over 10 classes. Sample images are shown in the first row of Figure 3.
- **MSRC-v1**² is a widely used data set for object recognition. This data set contains 38 different classes, with 30 images in each class. In the experiments, we select 7 classes as an experimental data set. Besides, we also extract four visual

¹<http://archive.ics.uci.edu/ml/datasets/Multiple+Features>

feature sets from each image. Sample images are shown in the second row of Figure 3.

- **ORL**³ is one of the best-known data sets in the face clustering literature. This data set contains a set of 400 face images, with 10 images in each subject. Particularly, these 10 various images are taken at different times, lighting, facial expressions, and facial details. Sample images are shown in the third row of Figure 3.
- **Yale**⁴ is a well-known face image data set that contains 165 grayscale images of 15 subjects. For each subject, there are 11 images with different facial expressions and configurations. Specially, there are three feature sets (i.e., views) in this data set. Sample images are shown in the fourth row of Figure 3.
- **YaleB**⁵ is a face image data set described by three views. In summary, there are 38 diverse subjects of face images in this data set. We select the first 10 subjects for multi-view clustering, with 65 images for each subject. Sample images are shown in the fifth row of Figure 3.
- **VIS/NIR** is a data set consisting of face images whose features are available in two different light environments (i.e., views): Visible Light and Near-IR illumination. Similar to the work in Zhang et al. (2020), we construct a subset containing 1056 images for multi-view clustering. Sample images are shown in the sixth row of Figure 3.
- **Reuters** is a data set that contains multilingual documents corresponding to six Reuters categories. Same to the previous work, we use a randomly sampled subset containing 600 documents in our experiments.
- **Cora** is a link-based document data set, which contains two types of views, i.e., the document contents and document citations. It consists totally of 1051

²<http://research.microsoft.com/en-us/projects/objectclassrecognition/>

³<http://www.uk.research.att.com/facedatabase.html>

⁴<http://vision.ucsd.edu/content/yale-face-database>

⁵<http://vision.ucsd.edu/~iskwak/ExtYaleDatabase/Yale%20Face%20Database.htm>



Figure 3: Sample images of used data sets.

305 machine learning papers, belonging to two classes.

In our experiments, three popular metrics are used to evaluate the clustering performance: Accuracy (Acc) Zhong & Ghosh (2003), Normalized Mutual Information (NMI) Huang et al. (2019a) and F-score Lin et al. (2018).

310 Considering two different partitions, let ν be the predicted partition of K clusters, and ω be the target partition of L classes. Given a data set with N samples, the NMI score between ν and ω is computed as follows:

$$\text{NMI}(\nu, \omega) = \frac{2 \sum_{k=1}^K \sum_{l=1}^L \frac{n_k^l}{N} \log \frac{n_k^l N}{\sum_{i=1}^K n_i^l \sum_{i=1}^L n_k^i}}{H(\nu) + H(\omega)}. \quad (26)$$

Here $H(\nu)$ and $H(\omega)$ denote the Shannon entropy of predicted partition ν and target partition ω respectively. Besides, here n_i and n^j are the numbers of data samples in cluster i and class j , and n_i^j represents the number of data samples in cluster i and class

315 j .

Acc is another important evaluation metric that measures the correct rate of predicted partition and the target partition. Suppose a data set X with N data samples, we denote ν as the predicted partition and ω as the target partition. Then the definition of Acc could be expressed as:

$$\text{Acc}(\nu, \omega) = \frac{\sum_{i=1}^N \delta(\pi, \text{map}(\omega))}{N}. \quad (27)$$

Here $\delta(\cdot)$ is the delta function, and $\text{map}(\cdot)$ is the best permutation mapping function which projects each predicted partition to the target partition.

Before calculating the value of F-score, we need to obtain Precision and Recall in advance. The values of the Precision and Recall metrics can be computed by:

$$\text{Precision}(C_\alpha, C_\beta) = \frac{N_{\alpha,\beta}}{|C_\alpha|}, \quad (28)$$

$$\text{Recall}(C_\alpha, C_\beta) = \frac{N_{\alpha,\beta}}{|C_\beta|}, \quad (29)$$

where C_α and C_β are two different clusters, and $N_{\alpha,\beta}$ represents the number of data samples from C_α in the class C_β . $|C_\alpha|$ and $|C_\beta|$ represent the module of the class C_α and C_β respectively.

Finally, we can compute the F-score between two clusters C_α and C_β by:

$$F(C_\alpha, C_\beta) = \frac{2\text{Precision}(C_\alpha, C_\beta)\text{Recall}(C_\alpha, C_\beta)}{\text{Precision}(C_\alpha, C_\beta) + \text{Recall}(C_\alpha, C_\beta)}. \quad (30)$$

These metrics characterize different properties of the clustering results, thus a comprehensive analysis could be acquired from multiple viewpoints. Particularly, for all three evaluations, the larger values indicate the better results.

5.2. Comparison Results and Analysis

In this subsection, we compare the proposed JRL-MSC with eleven state-of-the-art clustering methods for comprehensive comparisons. The compared methods include three single-view subspace clustering methods, three multi-view graph clustering methods and five multi-view subspace clustering methods. Particularly, for the single-view subspace clustering methods, we need to concatenate the features from different views

and feed them into these algorithms. Note that SMR has been already introduced in section 4, thus we will not introduce this method in this subsection. The brief introduction of the remaining methods is given as follows.

- 340 • **Sparse Subspace Clustering (SSC) Elhamifar & Vidal (2013):** SSC is a typical method in the single-view subspace clustering literature. The goal of SSC is to cluster the data samples into their subspaces with sparse self-expression property.
- 345 • **Low Rank Recovery Representation (LRR) Liu et al. (2012):** LRR is another representative method in the field of single-view subspace clustering. It aims to group the data samples into several subspaces with low-rank representation.
- **Multi-view Learning with Adaptive Neighbors (MLAN) Nie et al. (2017a):** MLAN is one of the most famous methods in the domain of multi-view graph clustering. To be specific, this method is capable to perform multi-view clustering and local structure learning simultaneously.
- 350 • **Self-weighted Multiview Clustering (SwMC) Nie et al. (2017b):** SwMC is a well-known method for multi-view graph clustering. In this method, the key idea is to generate a robust global graph by taking advantage of the complementarity across different views.
- 355 • **Graph-based Multi-view Clustering (GMC) Wang et al. (2020):** GMC is a novel multi-view graph clustering method, whose purpose is to adaptively reveal the underlying graph structure of multiple views under one-step framework.
- **Diversity-induced Multi-view Subspace Clustering (DiMSC) Cao et al. (2015):** DiMSC is a classical subspace clustering method for multi-view data, which explores the diversity of different views through HSIC regularizer.
- 360 • **Low-Rank Tensor Constrained Multiview Subspace Clustering (LT-MSC) Zhang et al. (2015):** LT-MSC is a tensor multi-view subspace clustering method, whose main idea is to make use of the high-order correlations by a low-rank tensor constraint.

- 365 • **Latent Multi-view Subspace Clustering (LMSC) Zhang et al. (2017a):** LM-SC is a novel multi-view subspace clustering method, which simultaneously learns the latent space from multiple views as well as discovers the subspace structure in the learned space. To the best of our knowledge, LMSC makes the first attempt to solve the view-insufficiency problem in multi-view clustering.
- 370 • **Consistent and Specific Multi-view Subspace Clustering (CSMSC) Luo et al. (2018):** CSMSC is a newly developed method for multi-view subspace clustering. The method targets at learning subspace representation from both consistent and diverse perspectives.
- 375 • **Multi-view Clustering in Latent Embedding Space (MCLES) Chen et al. (2020):** MCLES is a very recently proposed method for multi-view subspace clustering. The method integrates two important tasks into a unified problem, namely, latent space learning and subspace clustering.

For the comparative methods, we run the source codes released by the authors, while adjusting their parameters in a wider range than the suggestion of published paper.⁶ For the proposed method, we carefully tune the balancing parameters in the same range of (0,2]. The experiments were repeated 20 times for all methods, and the average values with standard deviations were reported as the results. For each experiment, the best result is marked in highlighted in bold font.

385 Tables 2, 3 and 4 report the comparison results in terms of Acc, NMI and F-score, respectively. Note that the MCLES method runs too long on the UCI digit data set, so we do not report the corresponding results here. In addition, due to the very sparse structure of Cora data set, we could not perform the SSC and DiMSC methods on this data set. The main findings arising from these tables are as follows.

⁶In our experiments, we tune the parameters of competitive methods within the same range of $\{0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1.0, 2.0, \dots, 10, 30, 50, 70, 90, 100, 500, 1000\}$. For SMR, MLAN, SwMC methods, we tune their parameter k which controls the number of k -nearest neighbors within the range of $\{5, 10, 15, \dots, 50, 60, 70, 80, 90, 100\}$.

Table 2: Comparison results in terms of NMI (%) on eight benchmark data sets

	UCI Digit	MSRC-v1	ORL	Yale	YaleB	VIS/NIR	Reuters	Cora
SSC	82.7 $\pm$ 3.22	58.5 $\pm$ 3.20	83.0 $\pm$ 1.53	63.1 $\pm$ 2.20	50.0 $\pm$ 1.51	90.1 $\pm$ 2.20	41.9 $\pm$ 1.63	N/A
LRR	83.6 $\pm$ 0.23	47.1 $\pm$ 0.82	83.8 $\pm$ 1.00	65.2 $\pm$ 1.23	63.3 $\pm$ 0.30	96.7 $\pm$ 0.74	34.5 $\pm$ 1.68	64.2 $\pm$ 0.00
SMR	83.4 $\pm$ 0.22	50.8 $\pm$ 3.73	83.6 $\pm$ 1.12	69.5 $\pm$ 1.10	57.2 $\pm$ 0.00	94.4 $\pm$ 0.73	38.6 $\pm$ 1.06	70.3 $\pm$ 0.00
MLAN	93.9 $\pm$ 0.00	75.9 $\pm$ 0.00	84.3 $\pm$ 0.32	70.5 $\pm$ 0.66	37.1 $\pm$ 0.00	56.2 $\pm$ 0.48	26.0 $\pm$ 1.43	12.5 $\pm$ 6.80
SwMC	91.9 $\pm$ 0.00	74.3 $\pm$ 0.00	86.9 $\pm$ 0.00	65.6 $\pm$ 0.13	50.6 $\pm$ 0.00	96.1 $\pm$ 0.50	23.0 $\pm$ 0.21	26.3 $\pm$ 0.00
GMC	90.4 $\pm$ 0.00	76.8 $\pm$ 0.00	87.1 $\pm$ 0.00	68.9 $\pm$ 0.00	44.9 $\pm$ 0.00	98.5 $\pm$ 0.00	24.9 $\pm$ 0.00	21.7 $\pm$ 0.00
DiMSC	77.2 $\pm$ 0.23	71.4 $\pm$ 0.22	91.8 $\pm$ 1.24	71.6 $\pm$ 1.51	63.5 $\pm$ 0.21	92.8 $\pm$ 1.72	30.4 $\pm$ 3.01	N/A
LT-MSC	85.7 $\pm$ 0.71	75.2 $\pm$ 0.52	92.1 $\pm$ 1.26	76.7 $\pm$ 0.44	63.7 $\pm$ 0.33	99.0 $\pm$ 0.30	30.8 $\pm$ 0.23	32.4 $\pm$ 0.00
LMSC	82.5 $\pm$ 2.01	67.4 $\pm$ 6.06	91.2 $\pm$ 1.62	76.8 $\pm$ 1.93	63.6 $\pm$ 1.68	98.8 $\pm$ 3.93	31.8 $\pm$ 3.01	72.4 $\pm$ 1.10
CSMSC	87.5 $\pm$ 0.00	77.5 $\pm$ 0.00	93.3 $\pm$ 0.79	78.7 $\pm$ 0.38	55.0 $\pm$ 0.00	95.3 $\pm$ 0.23	32.7 $\pm$ 0.00	59.8 $\pm$ 0.00
MCLES	N/A	79.8 $\pm$ 0.75	91.3 $\pm$ 0.98	72.0 $\pm$ 2.05	40.5 $\pm$ 0.83	85.2 $\pm$ 2.30	32.6 $\pm$ 2.33	67.9 $\pm$ 0.00
JRL-MSC	96.6 $\pm$ 0.00	100.0 $\pm$ 0.00	98.9 $\pm$ 0.31	93.4 $\pm$ 0.63	75.0 $\pm$ 0.00	100.0 $\pm$ 0.00	65.2 $\pm$ 0.00	76.2 $\pm$ 0.00

Table 3: Comparison results in terms of Acc (%) on eight benchmark data sets.

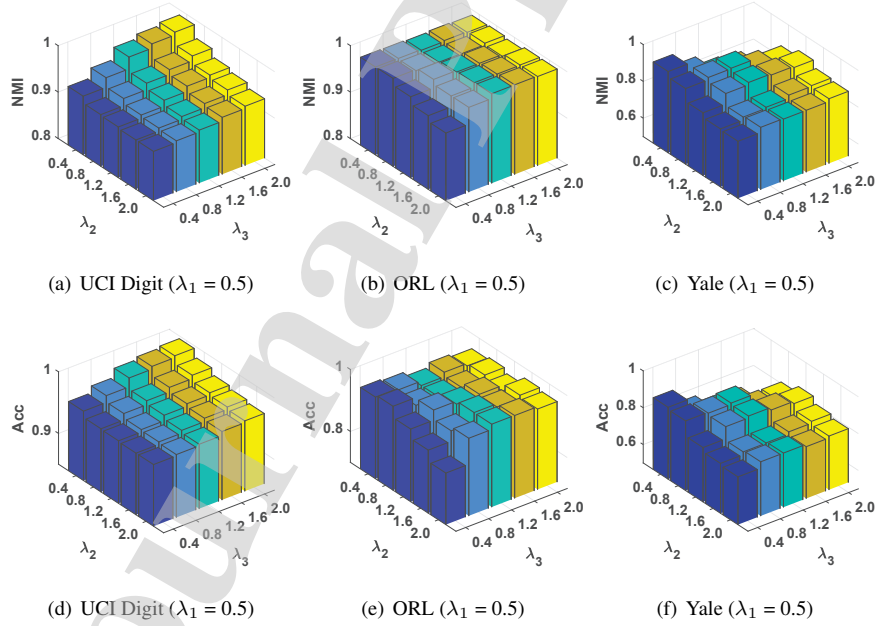
	UCI Digit	MSRC-v1	ORL	Yale	YaleB	VIS/NIR	Reuters	Cora
SSC	73.5 $\pm$ 4.02	68.1 $\pm$ 4.38	66.4 $\pm$ 3.00	60.4 $\pm$ 3.73	51.3 $\pm$ 3.83	69.7 $\pm$ 4.14	56.3 $\pm$ 1.49	N/A
LRR	90.1 $\pm$ 0.14	59.1 $\pm$ 0.84	69.5 $\pm$ 2.53	63.7 $\pm$ 2.30	65.0 $\pm$ 0.32	87.6 $\pm$ 2.83	47.1 $\pm$ 2.95	95.0 $\pm$ 0.00
SMR	81.7 $\pm$ 0.61	61.3 $\pm$ 4.84	68.7 $\pm$ 2.33	65.7 $\pm$ 0.92	56.2 $\pm$ 0.00	86.6 $\pm$ 2.51	47.7 $\pm$ 1.76	94.8 $\pm$ 0.00
MLAN	97.3 $\pm$ 0.00	74.2 $\pm$ 0.00	67.8 $\pm$ 1.51	67.6 $\pm$ 0.83	36.4 $\pm$ 0.00	36.7 $\pm$ 0.23	27.8 $\pm$ 1.52	77.6 $\pm$ 1.30
SwMC	89.3 $\pm$ 0.00	72.4 $\pm$ 0.00	68.3 $\pm$ 0.00	64.3 $\pm$ 0.22	50.2 $\pm$ 0.00	91.6 $\pm$ 0.93	25.8 $\pm$ 0.00	85.1 $\pm$ 0.20
GMC	88.2 $\pm$ 0.00	73.4 $\pm$ 0.00	65.3 $\pm$ 0.00	65.5 $\pm$ 0.00	43.4 $\pm$ 0.00	95.5 $\pm$ 0.00	30.3 $\pm$ 0.00	80.6 $\pm$ 0.00
DiMSC	70.3 $\pm$ 0.31	80.7 $\pm$ 0.33	81.0 $\pm$ 2.14	69.6 $\pm$ 2.09	61.5 $\pm$ 0.32	90.6 $\pm$ 3.12	45.7 $\pm$ 0.30	N/A
LT-MSC	92.3 $\pm$ 0.63	82.4 $\pm$ 0.33	81.3 $\pm$ 2.08	73.6 $\pm$ 0.28	62.6 $\pm$ 1.03	97.2 $\pm$ 3.00	47.7 $\pm$ 0.27	82.7 $\pm$ 0.00
LMSC	84.0 $\pm$ 5.80	72.3 $\pm$ 5.90	80.5 $\pm$ 1.72	74.2 $\pm$ 2.44	62.4 $\pm$ 2.14	97.0 $\pm$ 0.56	45.3 $\pm$ 5.04	95.8 $\pm$ 2.43
CSMSC	93.5 $\pm$ 0.00	85.2 $\pm$ 0.00	85.2 $\pm$ 1.75	75.5 $\pm$ 0.97	54.2 $\pm$ 0.00	88.6 $\pm$ 0.45	47.9 $\pm$ 0.00	94.1 $\pm$ 0.00
MCLES	N/A	88.2 $\pm$ 0.38	79.5 $\pm$ 2.18	69.3 $\pm$ 2.07	42.7 $\pm$ 0.50	74.3 $\pm$ 1.99	42.1 $\pm$ 0.00	95.7 $\pm$ 0.00
JRL-MSC	98.5 $\pm$ 0.00	100.0 $\pm$ 0.00	95.6 $\pm$ 1.38	90.3 $\pm$ 0.83	70.9 $\pm$ 0.00	100.0 $\pm$ 0.00	72.8 $\pm$ 0.00	96.8 $\pm$ 0.00

- First, it can be seen that single-view subspace methods sometimes show comparable performance to the multi-view subspace methods, but still have gaps on the most of the data sets. This is probably for the reason that single-view subspace methods directly concatenate the multi-view features, which may lead to the redundant information in most of cases.
- Second, for the multi-view clustering cases, we can observe that subspace-based methods always achieve more promising results than the graph-based methods (except on the UCI Digit data set). This indicates that subspace-based methods

Table 4: Comparison results in terms of F-score (%) on eight benchmark data sets.

	UCI Digit	MSRC-v1	ORL	Yale	YaleB	VIS/NIR	Reuters	Cora
SSC	68.5 $\pm$ 6.80	53.6 $\pm$ 3.58	52.3 $\pm$ 3.70	41.0 $\pm$ 2.55	32.3 $\pm$ 0.93	70.9 $\pm$ 5.37	44.3 $\pm$ 0.43	N/A
LRR	79.2 $\pm$ 0.23	45.4 $\pm$ 0.86	55.8 $\pm$ 1.88	44.2 $\pm$ 1.44	39.0 $\pm$ 0.42	89.6 $\pm$ 2.01	38.6 $\pm$ 0.56	92.9 $\pm$ 0.00
SMR	77.0 $\pm$ 2.04	47.6 $\pm$ 3.30	56.7 $\pm$ 2.64	50.6 $\pm$ 1.40	40.8 $\pm$ 0.08	86.9 $\pm$ 1.19	40.8 $\pm$ 0.31	92.3 $\pm$ 0.00
MLAN	93.8 $\pm$ 0.00	67.2 $\pm$ 0.00	38.5 $\pm$ 1.24	52.1 $\pm$ 1.30	24.2 $\pm$ 0.00	27.7 $\pm$ 5.09	29.3 $\pm$ 0.48	78.4 $\pm$ 0.21
SwMC	87.5 $\pm$ 0.00	66.9 $\pm$ 0.00	39.7 $\pm$ 0.00	47.3 $\pm$ 0.02	32.9 $\pm$ 0.00	91.2 $\pm$ 5.13	28.9 $\pm$ 0.10	80.7 $\pm$ 0.00
GMC	86.5 $\pm$ 0.00	68.5 $\pm$ 0.00	53.9 $\pm$ 0.00	48.0 $\pm$ 0.00	28.5 $\pm$ 0.00	94.7 $\pm$ 0.00	32.2 $\pm$ 0.00	81.7 $\pm$ 0.00
DiMSC	69.5 $\pm$ 0.09	68.7 $\pm$ 0.35	76.8 $\pm$ 2.17	54.6 $\pm$ 2.26	50.4 $\pm$ 0.58	89.4 $\pm$ 3.05	35.1 $\pm$ 0.23	N/A
LT-MSC	85.0 $\pm$ 1.03	71.8 $\pm$ 0.36	75.7 $\pm$ 2.34	61.2 $\pm$ 0.65	52.1 $\pm$ 0.57	96.8 $\pm$ 1.93	37.3 $\pm$ 0.11	83.4 $\pm$ 0.00
LMSC	78.2 $\pm$ 3.50	63.5 $\pm$ 8.23	73.9 $\pm$ 2.14	57.3 $\pm$ 1.77	52.8 $\pm$ 1.40	93.2 $\pm$ 7.49	35.7 $\pm$ 2.14	93.8 $\pm$ 3.52
CSMSC	86.5 $\pm$ 0.00	75.8 $\pm$ 0.00	78.1 $\pm$ 2.00	64.2 $\pm$ 0.57	36.8 $\pm$ 0.00	86.8 $\pm$ 1.00	37.5 $\pm$ 0.00	91.7 $\pm$ 0.00
MCLES	N/A	79.0 $\pm$ 0.60	71.3 $\pm$ 0.40	53.8 $\pm$ 2.96	28.7 $\pm$ 0.54	70.2 $\pm$ 3.43	37.1 $\pm$ 0.73	93.8 $\pm$ 0.00
JRL-MSC	97.0 $\pm$ 0.00	100.0 $\pm$ 0.00	95.5 $\pm$ 1.45	86.1 $\pm$ 1.07	66.4 $\pm$ 0.00	100.0 $\pm$ 0.00	63.0 $\pm$ 0.00	94.6 $\pm$ 0.00

may be more applicable to solve the image clustering problems in practice.

Figure 4: Analysis on the effect of parameters λ_2 and λ_3 by fixing parameter λ_1 .

- Finally, results indicate that the proposed method consistently outperforms the compared methods on all the tested data sets. Surprisingly, our method gains significant improvement over the second-best method. Take Yale data set as an example, our method obtains an increase of more than 14.9%, 13.9 % and 21.9 % over the NMI, ACC and F-score metrics, compared with the other competitors. This is mainly due to the fact that our method can jointly preserve the view-specific underlying structure within each view, and discover the high-order correlation among diverse views. In multi-view subspace clustering domain, these two components are essential to enhance the clustering performance. To sum up, these comprehensive results validly prove the superior performance and the robustness of the proposed method.

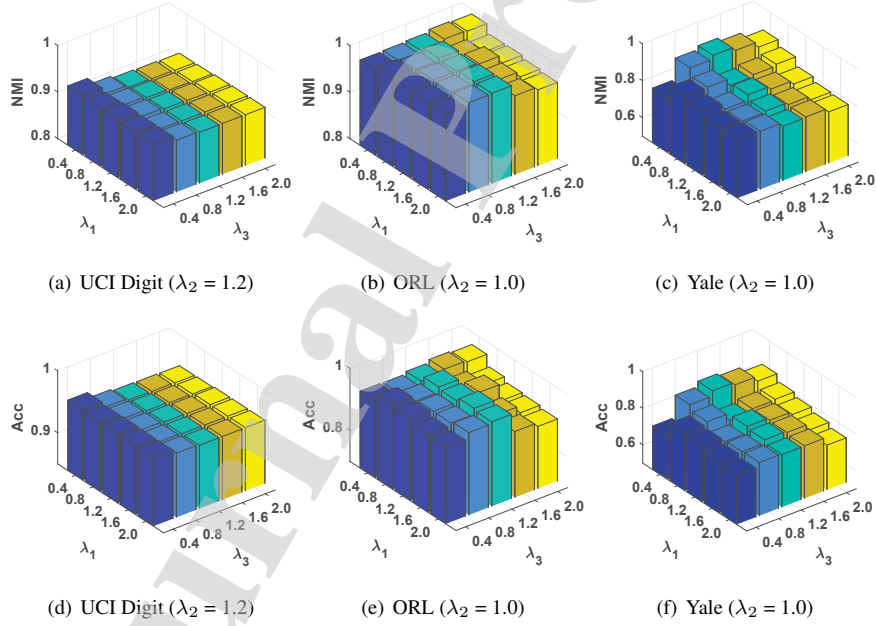


Figure 5: Analysis on the effect of parameters λ_1 and λ_3 by fixing parameter λ_2 .

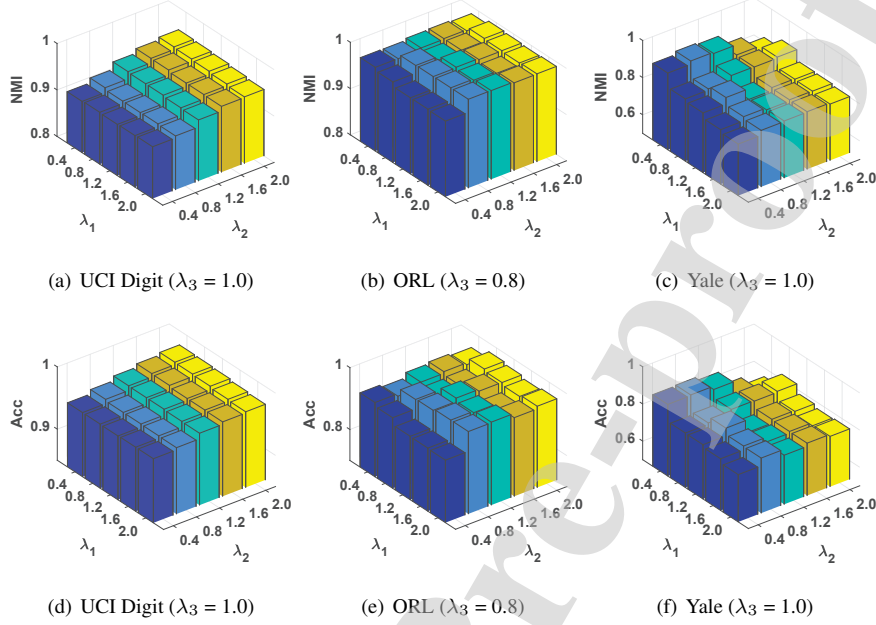


Figure 6: Analysis on the effect of parameters λ_1 and λ_2 by fixing parameter λ_3 .

5.3. Parameter Analysis and Running Time Analysis

In this subsection, we investigate the effects of different parameters (i.e., λ_1 , λ_2 and λ_3) on the performance of our method. Particularly, we select the values of λ_1 , λ_2 and λ_3 from the same predefined range of $\{0.4, 0.8, 1.2, 1.6, 2.0\}$. Moreover, we analyze the sensitivity of any two parameters, while keeping the rest one in each experiment fixed. In this experiment, three data sets are selected for the parameter analysis. Meanwhile, the clustering results are evaluated in metrics of NMI and Acc.

The clustering results under different parameter settings are presented in Figure 4 to Figure 6. Based on the results in these figures, it can be observed that our method performs fairly stable with different values of λ_1 , λ_2 and λ_3 , especially on the UCI Digit and ORL data sets. Furthermore, we can find that the proposed method obtains the best performance when the parameters λ_1 , λ_2 and λ_3 are tuned in the range from 0.4 to 1.2. The experimental results imply that our method is relatively robust for the

parameters that change in different clustering tasks.

After that, we report the running time of our method as well as the comparative
 425 methods on six multi-view data sets. From the results shown in table 5, we can clearly see that multi-view graph methods run faster over the other comparison methods, including single-view and multi-view subspace methods. When it comes to the multi-view subspace methods, we also find that the proposed method is more efficient than the competitors in most of the cases. In summary, the running time of our method ranks
 430 at the middle-upper level among all the clustering methods.

Table 5: Running time (in seconds) of all clustering methods on eight benchmark data sets.

Method	UCI Digit	MSRC-v1	ORL	Yale	YaleB	VIS/NIR	Reuters	Cora
SSC	55.97	1.03	25.20	3.60	47.70	190.84	6.54	N/A
LRR	581.41	3.51	42.91	11.85	86.59	292.90	110.68	138.64
SMR	30.45	0.06	1.25	0.33	1.89	7.53	0.48	4.82
MLAN	15.65	0.10	1.18	0.22	2.67	5.27	51.04	1.54
SwMC	99.43	0.49	0.85	3.31	7.65	34.55	14.11	20.97
GMC	13.79	0.22	0.94	0.32	1.38	2.52	66.87	1.16
DiMSC	2754.23	1.85	10.03	1.52	30.52	80.87	37.47	N/A
LT-MSC	984.17	4.30	56.88	16.15	111.92	276.14	131.52	127.21
LMSC	481.54	3.86	40.35	28.72	80.05	128.80	75.24	102.98
CSMSC	473.96	2.92	72.41	24.39	138.38	404.63	16.41	159.23
MCLES	N/A	5.83	35.27	5.67	243.74	3610.21	165.30	1214.06
JRL-MSC	1650.84	1.51	7.85	1.52	23.76	75.40	8.59	71.61

5.4. Ablation Study

In this subsection, we conduct ablation study by comparing our method with two baselines that are special cases of the proposed method. The brief introduction of these two baselines are as follows:

- 435 • **SMR based Multi-view Subspace Clustering (SMR-MSC)** : SMR-MSC is a special case of our method which only considers the view-specific local structure within each view. The objective function of SMR-MSC can be found in Eq. (12).
- **Tensor Representation Learning for Multi-view Subspace Clustering (TRL-MSC)**: TRL-MSC is another special case of the proposed method which merely

Table 6: Ablation study in terms of NMI (%) on eight benchmark data sets.

	UCI Digit	MSRC-v1	ORL	Yale	YaleB	VIS/NIR	Reuters	Cora
SMR-MSC	91.0 $\pm$ 0.00	70.1 $\pm$ 0.00	93.5 $\pm$ 0.47	76.8 $\pm$ 1.24	54.8 $\pm$ 0.00	98.6 $\pm$ 0.40	40.3 $\pm$ 0.26	67.9 $\pm$ 0.00
TRL-MSC	84.7 $\pm$ 0.26	92.8 $\pm$ 0.00	97.7 $\pm$ 0.63	87.5 $\pm$ 1.66	65.2 $\pm$ 0.10	98.8 $\pm$ 0.17	59.4 $\pm$ 0.00	73.6 $\pm$ 0.00
JRL-MSC	96.6 $\pm$ 0.00	100.0 $\pm$ 0.00	98.9 $\pm$ 0.31	93.4 $\pm$ 0.63	75.0 $\pm$ 0.00	100.0 $\pm$ 0.00	65.2 $\pm$ 0.00	76.2 $\pm$ 0.00

Table 7: Ablation study in terms of Acc (%) on eight benchmark data sets.

	UCI Digit	MSRC-v1	ORL	Yale	YaleB	VIS/NIR	Reuters	Cora
SMR-MSC	95.7 $\pm$ 0.00	83.3 $\pm$ 0.00	83.8 $\pm$ 1.50	75.4 $\pm$ 1.67	54.0 $\pm$ 0.00	94.8 $\pm$ 1.47	52.8 $\pm$ 0.31	95.2 $\pm$ 0.00
TRL-MSC	91.7 $\pm$ 0.21	95.2 $\pm$ 0.00	93.5 $\pm$ 1.94	89.1 $\pm$ 1.10	63.4 $\pm$ 0.10	98.9 $\pm$ 0.14	65.7 $\pm$ 0.00	96.2 $\pm$ 0.00
JRL-MSC	98.5 $\pm$ 0.00	100.0 $\pm$ 0.00	95.6 $\pm$ 1.38	90.3 $\pm$ 0.83	70.9 $\pm$ 0.00	100.0 $\pm$ 0.00	72.8 $\pm$ 0.00	96.8 $\pm$ 0.00

Table 8: Ablation study in terms of F-score (%) on eight benchmark data sets.

	UCI Digit	MSRC-v1	ORL	Yale	YaleB	VIS/NIR	Reuters	Cora
SMR-MSC	91.7 $\pm$ 0.00	70.6 $\pm$ 0.00	77.2 $\pm$ 1.29	58.4 $\pm$ 1.41	37.3 $\pm$ 0.00	95.1 $\pm$ 1.33	42.6 $\pm$ 0.16	93.4 $\pm$ 0.00
TRL-MSC	82.7 $\pm$ 0.37	91.2 $\pm$ 0.00	92.9 $\pm$ 2.01	79.8 $\pm$ 2.08	52.6 $\pm$ 0.22	97.8 $\pm$ 0.27	56.3 $\pm$ 0.00	94.3 $\pm$ 0.00
JRL-MSC	97.0 $\pm$ 0.00	100.0 $\pm$ 0.00	95.5 $\pm$ 1.45	86.1 $\pm$ 1.07	66.4 $\pm$ 0.00	100.0 $\pm$ 0.00	63.0 $\pm$ 0.00	94.6 $\pm$ 0.00

considers the higher correlation across different views. The objective function of TRL-MSC can be found in Eq. (13).

The clustering results in terms of NMI, Acc and F-score are shown in Tables 6, 7 and 8, respectively. According to the results reported in these tables, we can observe two aspects of findings. First, TRL-MSC substantially outperforms the SMR-MSC on seven of eight tested data sets (except on the UCI Digit data set). This indicates that efficiently exploring the high-order correlations makes greater contribution to the clustering performance. Second, the proposed methods JRL-MSC performs better than two baselines (i.e., SMR-MSC and TRL-MSC) on all the tested data sets. These two baselines belong to the special cases of our method, which only consider one component of JRL-MSC and have their merits. Therefore, the better way to enhance the clustering performance relies on considering the local structure within each view and the high-order complementarity across different views jointly.

5.5. Convergence Analysis

In this subsection, we empirically analyze the convergence property of our method on four tested data sets, namely HW, MSRC-v1, Yale and YaleB. In particular, the convergence condition of our method is determined by the reconstruction error (i.e., $\sum_{v=1}^V \|X^{(v)} - X^{(v)}Z^{(v)} - E^{(v)}\|_{\infty}$), as shown in Eq. (25). Based on the curves displayed in Figure 7, we can see that the values of reconstruction error decrease rapidly during the iteration. On all the tested data sets, the proposed method tends to converge within a small number of iteration (i.e., around 10 to 20 iterations). This result confirms that our method JRL-MSC is relatively efficient for handling multi-view subspace clustering problem.

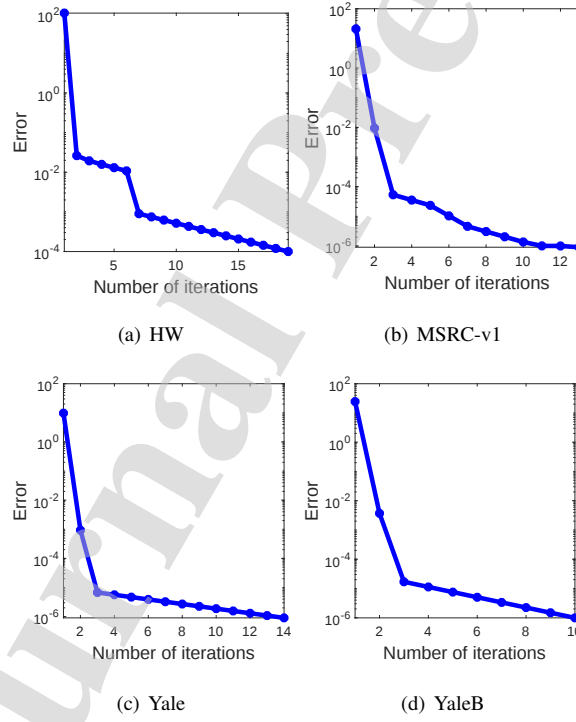


Figure 7: Convergence analysis of JRL-MSC on four benchmark data sets.

6. Conclusion and Future Work

In this paper, we propose a novel multi-view subspace clustering based on the joint representation learning framework. Notably, the proposed method aims to learn the view-specific representation by discovering the local geometrical structure within each view. In the meanwhile, our method targets at generating the low-rank tensor representation for capturing the global complementarity across different views. These two representation learning processes are jointly leveraged into our method, which can be regarded as the local and global constraints respectively. Compared with the existing methods, the proposed method is able to explore the intra-view pairwise relationship as well as the intra-view high-order correlation for boosting the performance of multi-view subspace clustering. Finally, comprehensive experiments on different multi-view data sets demonstrated the effectiveness of our method. In the future work, we would like to combine our method with representation learning of deep learning, which aims to handle the more challenging multi-view clustering problems in practice.

Acknowledgments

The authors would like to thank the anonymous reviewers and the Editor for their constructive comments and suggestions, which greatly improve this paper. This work was supported by the NSFC (61773410, 61673403, 61976097) and Guangdong Basic and Applied Basic Research Foundation (2019A151511154).

References

- Bartels, R. H., & Stewart, G. W. (1972). Solution of the matrix equation $ax + xb = c$. *Communications of the ACM*, 15, 820–826.
- Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J. et al. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3, 1–122.
- Brbić, M., & Kopriva, I. (2018). Multi-view low-rank sparse subspace clustering. *Pattern Recognition*, 73, 247–258.

- 490 Cai, X., Nie, F., & Huang, H. (2013). Multi-view k-means clustering on big data. In *Proceedings of the Twenty-Third International Joint conference on artificial intelligence* (pp. 2598–2604).
- Cao, X., Zhang, C., Fu, H., Liu, S., & Zhang, H. (2015). Diversity-induced multi-view subspace clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 586–594).
495
- Chen, M., Huang, L., Wang, C., & Huang, D. (2020). Multi-view clustering in latent embedding space. In *Thirty-Fourth AAAI Conference on Artificial Intelligence* (pp. 3513–3520).
- Chen, X., Xu, X., Huang, J. Z., & Ye, Y. (2013). Tw-k-means: automated two-level
500 variable weighting clustering algorithm for multiview data. *IEEE Transactions on Knowledge and Data Engineering*, 25, 932–944.
- Elhamifar, E., & Vidal, R. (2013). Sparse subspace clustering: Algorithm, theory, and applications. *IEEE transactions on pattern analysis and machine intelligence*, 35, 2765–2781.
- 505 Gao, H., Nie, F., Li, X., & Huang, H. (2015). Multi-view subspace clustering. In *Proceedings of the IEEE international conference on computer vision* (pp. 4238–4246).
- Heckel, R., & Bölcskei, H. (2015). Robust subspace clustering via thresholding. *IEEE Transactions on Information Theory*, 61, 6320–6342.
- 510 Hu, H., Lin, Z., Feng, J., & Zhou, J. (2014). Smooth representation clustering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3834–3841).
- Huang, D., Wang, C., Wu, J., Lai, J., & Kwok, C. K. (2019a). Ultra-scalable spectral clustering and ensemble clustering. *IEEE Transactions on Knowledge and Data Engineering*, 32, 1212–1226.
515

- Huang, L., Chao, H., & Wang, C. (2019b). Multi-view intact space clustering. *Pattern Recognition*, 86, 344–353.
- Huang, S., Kang, Z., Tsang, I. W., & Xu, Z. (2019c). Auto-weighted multi-view clustering via kernelized graph learning. *Pattern Recognition*, 88, 174–184.
- 520 Jiang, Y., Chung, F., Wang, S., Deng, Z., Wang, J., & Qian, P. (2014). Collaborative fuzzy clustering from multiple weighted views. *IEEE transactions on cybernetics*, 45, 688–701.
- Kilmer, M. E., Braman, K., Hao, N., & Hoover, R. C. (2013). Third-order tensors as operators on matrices: A theoretical and computational framework with applications
525 in imaging. *SIAM Journal on Matrix Analysis and Applications*, 34, 148–172.
- Kilmer, M. E., & Martin, C. D. (2011). Factorization strategies for third-order tensors. *Linear Algebra and its Applications*, 435, 641–658.
- Lane, C., Haeffele, B., & Vidal, R. (2019). Adaptive online k-subspaces with cooperative re-initialization. In *Proceedings of the IEEE International Conference on
530 Computer Vision Workshops* (pp. 678–688).
- Li, J., Wang, C., Li, P., & Lai, J. (2018). Discriminative metric learning for multi-view graph partitioning. *Pattern Recognition*, 75, 199–213.
- Li, Y., Nie, F., Huang, H., & Huang, J. (2015). Large-scale multi-view spectral clustering via bipartite graph. In *Proceedings of the Twenty-Ninth AAAI Conference on
535 Artificial Intelligence* (pp. 2750–2756).
- Liang, Y., Huang, D., & Wang, C. (2019). Consistency meets inconsistency: A unified graph learning framework for multi-view clustering. In *Proceedings of the IEEE International Conference on Data Mining* (pp. 1204–1209).
- Lin, K., Huang, L., Wang, C., & Chao, H. (2018). Multi-view proximity learning for
540 clustering. In *Proceedings of the International Conference on Database Systems for Advanced Applications* (pp. 407–423).

- Lipor, J., Hong, D., Tan, Y. S., & Balzano, L. (2017). Subspace clustering using ensembles of k -subspaces. *arXiv preprint arXiv:1709.04744*, .
- 545 Liu, G., Lin, Z., Yan, S., Sun, J., Yu, Y., & Ma, Y. (2012). Robust recovery of subspace structures by low-rank representation. *IEEE transactions on pattern analysis and machine intelligence*, 35, 171–184.
- Liu, G., Lin, Z., Yan, S., Sun, J., Yu, Y., & Ma, Y. (2013a). Robust recovery of subspace structures by low-rank representation. *IEEE transactions on pattern analysis and machine intelligence*, 35, 171–184.
- 550 Liu, J., Wang, C., Gao, J., & Han, J. (2013b). Multi-view clustering via joint nonnegative matrix factorization. In *Proceedings of the SIAM International Conference on Data Mining* (pp. 252–260).
- Luo, S., Zhang, C., Zhang, W., & Cao, X. (2018). Consistent and specific multi-view subspace clustering. In *Thirty-Second AAAI Conference on Artificial Intelligence* (pp. 3730–3737).
555
- Nie, F., Cai, G., & Li, X. (2017a). Multi-view clustering and semi-supervised classification with adaptive neighbours. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence* (pp. 2408–2414).
- Nie, F., Li, J., & Li, X. (2017b). Self-weighted multiview clustering with multiple graphs. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence* (pp. 2564–2570).
560
- Nie, F., Shi, S., & Li, X. (2020). Auto-weighted multi-view co-clustering via fast matrix factorization. *Pattern Recognit.*, 102, 107207.
- Park, D., Caramanis, C., & Sanghavi, S. (2014). Greedy subspace clustering. In *Advances in Neural Information Processing Systems* (pp. 2753–2761).
565
- Rahmani, M., & Atia, G. K. (2017). Innovation pursuit: A new approach to subspace clustering. *IEEE Transactions on Signal Processing*, 65, 6276–6291.

- Tzortzis, G., & Likas, A. (2012). Kernel-based weighted multi-view clustering. In *Proceedings of the IEEE International Conference on Data Mining* (pp. 675–684).
- 570 Wang, H., Yang, Y., & Liu, B. (2020). Gmc: Graph-based multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 32, 1116–1129.
- Wang, X., Guo, X., Lei, Z., Zhang, C., & Li, S. Z. (2017). Exclusivity-consistency regularized multi-view subspace clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 923–931).
- 575 Wang, Y., Lin, X., Wu, L., Zhang, W., Zhang, Q., & Huang, X. (2015). Robust subspace clustering for multi-view data by exploiting correlation consensus. *IEEE Transactions on Image Processing*, 24, 3939–3949.
- Wang, Y., Xu, H., & Leng, C. (2013). Provable subspace clustering: When lrr meets ssc. In *Advances in Neural Information Processing Systems* (pp. 64–72).
- 580 Wu, J., Lin, Z., & Zha, H. (2019). Essential tensor learning for multi-view spectral clustering. *IEEE Transactions on Image Processing*, 28, 5910–5922.
- Wu, J., Xie, X., Nie, L., Lin, Z., & Zha, H. (2020). Unified graph and low-rank tensor learning for multi-view clustering. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI* (pp. 6388–6395).
- 585 Xie, Y., Tao, D., Zhang, W., Liu, Y., Zhang, L., & Qu, Y. (2018). On unifying multi-view self-representations for clustering by tensor multi-rank minimization. *International Journal of Computer Vision*, 126, 1157–1179.
- Xu, C., Tao, D., & Xu, C. (2015). Multi-view self-paced learning for clustering. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intel-*
590 *ligence* (pp. 3974–3980).
- Yang, J., Yin, W., Zhang, Y., & Wang, Y. (2009). A fast algorithm for edge-preserving variational multichannel image restoration. *SIAM J. Imaging Sciences*, 2, 569–592.

- Yao, S., Yu, G., Wang, J., Domeniconi, C., & Zhang, X. (2019). Multi-view multiple clustering. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence* (pp. 4121–4127). 595
- Yin, M., Gao, J., Xie, S., & Guo, Y. (2018). Multiview subspace clustering via tensorial t-product representation. *IEEE transactions on neural networks and learning systems*, 30, 851–864.
- Zhan, K., Niu, C., Chen, C., Nie, F., Zhang, C., & Yang, Y. (2018). Graph structure fusion for multiview clustering. *IEEE Transactions on Knowledge and Data Engineering*, 31, 1984–1993. 600
- Zhan, K., Zhang, C., Guan, J., & Wang, J. (2017). Graph learning for multiview clustering. *IEEE transactions on cybernetics*, (pp. 1–9).
- Zhang, C., Fu, H., Liu, S., Liu, G., & Cao, X. (2015). Low-rank tensor constrained multiview subspace clustering. In *Proceedings of the IEEE international conference on computer vision* (pp. 1582–1590). 605
- Zhang, C., Hu, Q., Fu, H., Zhu, P., & Cao, X. (2017a). Latent multi-view subspace clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4279–4287).
- Zhang, G., Wang, C., Huang, D., & Zheng, W. (2017b). Multi-view collaborative locally adaptive clustering with minkowski metric. *Expert Systems with Applications*, 86, 307–320. 610
- Zhang, G., Wang, C., Huang, D., Zheng, W., & Zhou, Y. (2018). Tw-co-k-means: Two-level weighted collaborative k-means for multi-view clustering. *Knowledge-Based Systems*, 150, 127–138. 615
- Zhang, G., Zhou, Y., He, X., Wang, C., & Huang, D. (2020). One-step kernel multi-view subspace clustering. *Knowledge-Based Systems*, 189, 105126.
- Zhao, H., Ding, Z., & Fu, Y. (2017). Multi-view clustering via deep matrix factorization. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence* (pp. 2921–2927). 620

Zhong, S., & Ghosh, J. (2003). A unified framework for model-based clustering. *Journal of machine learning research*, 4, 1001–1037.

Zhu, X., Zhang, S., Hu, R., He, W., Lei, C., & Zhu, P. (2018). One-step multi-view spectral clustering. *IEEE Transactions on Knowledge and Data Engineering*, 31, 2022–2034.

Zong, L., Zhang, X., Zhao, L., Yu, H., & Zhao, Q. (2017). Multi-view clustering via multi-manifold regularized non-negative matrix factorization. *Neural Networks*, 88, 74–89.

Highlights

1. This paper proposes a new multi-view subspace clustering method.
2. Our method is based on a joint representation learning framework.
3. Our method can preserve the view-specific information within each view.
4. Our method can capture the complementary information across multiple views.
5. Extensive experiments confirm the effectiveness of our method.

Credit author statement

Guang-Yu Zhang: Methodology, Software, Writing-Original draft, Software, Validation, Visualization, Data curation.

Yu-Ren Zhou: Supervision, Conceptualization, Funding acquisition, Project administration.

Chang-Dong Wang: Formal analysis, Conceptualization.

Dong Huang: Writing-Reviewing and Editing, Formal analysis.

Xiao-Yu He: Writing-Reviewing and Editing, Formal analysis.

Declaration of Interest Statement

The manuscript, or part of it, has neither been published nor is currently under consideration for publication by any other journal. All authors have read and approved the content. No disclosure, ethical or legal conflicts and no competing interests exist in this study. For the manuscript, there is not any potential financial or substantive conflict. We hope that it is in accordance with your publishing regulations and under consideration for publication once got approved.

Guang-Yu Zhang

Yu-Ren Zhou

Chang-Dong Wang

Dong Huang

Xiao-Yu He