Unified Low-rank Tensor Learning and Spectral Embedding for Multi-view Subspace Clustering

Lele Fu, Zhaoliang Chen, Yongyong Chen, and Shiping Wang

***Abstract*—Multi-view subspace clustering aims to utilize the comprehensive information of multi-source features to aggregate data into multiple subspaces. Recently, low-rank tensor learning has been applied to multi-view subspace clustering, which explores high-order correlations of multi-view data and has achieved remarkable results. However, these existing methods have certain limitations: 1) The learning processes of low-rank tensor and label indicator matrix are independent. 2) Variable contributions of different views to the consistent clustering results are not discriminated. To handle these issues, we propose a unified framework that integrates low-rank tensor learning and spectral embedding (ULTLSE) for multi-view subspace clustering. Specifically, the proposed model adopts the tensor singular value decomposition (t-SVD) based tensor nuclear norm to encode the low-rank property of the self-representation tensor, and a label indicator matrix via spectral embedding is simultaneously exploited. To distinguish the importance of various views, we learn a quantifiable weighting coefficient for each view. An effective recursion optimization algorithm is also developed to address the proposed model. Finally, we conduct comprehensive experiments on eight real-world datasets with three categories. The experimental results indicate that the proposed ULTLSE is advanced over existing state-of-the-art clustering methods.**

***Index Terms*—Multi-view subspace clustering, spectral embedding, low-rank tensor, t-SVD.**

I. Introduction

Subspace clustering [1], [2] intends to allocate data points from various clusters into corresponding subspaces, and each data point can be fitted with a linear combination of the remaining sample points attributed to the same subspace. Due to its encouraging performance, subspace clustering has been applied in many fields such as data dimensionality reduction [3] and pattern recognition [4]. In the past decades, multi-view data has gradually boomed with the development of multimedia technology. Intuitively, a view can be understood as a feature representation of objects, while multi-view refers to the representations of objects from multiple features. For instance, an image can be characterized in terms of color, texture, shape, etc. Therefore, fully inquiring into the information complementarity and consistency among multiple views is beneficial for more essential description of objects, thus promoting the clustering effects. Conventional single-view subspace clustering cannot efficiently handle multi-view data, so abundant multi-view subspace clustering algorithms emerge as the times require. What is more gratifying is that multi-view clustering algorithms have served a large number of application scenarios, including computer vision [5], [6], [7], disease prediction [8], [9], [10], and natural language processing [11], [12], [13].

Inspired by sparse subspace clustering (SSC) [14] and low-rank representation (LRR) [15], extensive multi-view clustering methods based on self-representation subspace learning are proposed. Luo et al. [16] explored both the specificity and consistency of subspace representations. Zhang et al. [5] learned the unique latent representation of multi-view data, from which a subspace representation was exploited. Kang et al. [17] chose a small amount of anchor samples to build a subgraph for each view, then an efficient method was proposed to integrate these subgraphs. Yang et al. [18] designed a multiplicative decomposition scheme for maintaining the structural consistency of all extracted constituents, which promoted the performance of variable splitting scheme. These above works have achieved the promising effects, but they all mine the internal correlations among multiple views at the matrix level. For multi-view data, it is more reasonable to refine discriminative and consistent data representation from the aspect of tensor.

Tensor-based multi-view subspace learning usually first combines the representation matrix of each view into a 3-order tensor, then it restores a low-rank tensor via a certain tensor nuclear norm. Further, a linear fusion method is used over the recovered low-rank tensor to obtain a consensus affinity matrix, which is fed into the spectral algorithm to yield clustering results. There are some representative tensor-oriented multi-view subspace clustering methods. For instance, Zhang et al. [19] adopted the sum of nuclear norms (SNN) to encode a low-rank tensor space, where SNN refers to the sum of the rank of each self-representation matrix. Xie et al. [20] employed the t-SVD based tensor nuclear norm [21] to minimize the rank of target tensor. Considering different contributions of varying singular values, a weighted version of t-SVD based tensor nuclear norm was developed in [22]. In addition to the application of self-representation in tensor-oriented multi-view learning, some other construction manners of affinity matrices have also been adopted. Wu et al. [23] constructed transition probability matrices for all views, which are assembled into an original tensor. Chen et al. [24] paid attention to the nonlinear structures in multi-view data and used

This work was partially supported by the National Natural Science Foundation of China under Grant Nos. U21A20472 and 62106063, and by the Shenzhen College Stability Support Plan under Grant No. GXWD20201230155427003-20200824113231001.

Lele Fu, Zhaoliang Chen, and Shiping Wang are with the College of Computer and Data Science, Fuzhou University, Fuzhou 350116, China and also with the Key Laboratory of Network Computing and Intelligent Information Processing, Fuzhou University, Fuzhou 350116, China (email: lawrencefzu@gmail.com, chenzl23@outlook.com, shipingwangphd@163.com). Yongyong Chen is with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen 518055, China (Email: YongyongChen.cn@gmail.com). Corresponding author: Shiping Wang.

a kernel-induced mapping to obtain the data representation of each view. Tensor-based methods have made new progress in exploring the correlations between views, but most of them still suffer from certain limitations: 1) The low-rank tensor and label indicator matrix are separately learned, which overlooks the dependence between them and may make it incapable to gain the optimal solution of the latter. 2) The final affinity matrix is obtained by linearly adding each slice of the low-rank tensor such as in [19], [20], [22], [23], which is neither short of interpretability nor consideration of the inherent differences in views.

In this paper, we propose a unified framework for multi-view subspace clustering that fuses low-rank tensor learning and spectral embedding. In the proposed method, the original tensor is constructed via reorganizing the self-representation matrices of all views, which contain the potential correlations of samples and is more robust than the similarity matrices constructed by Gaussian kernel. Then the t-SVD based tensor nuclear norm is used to characterize the low-rank properties of the target tensor. When optimizing the low-rank tensor, a low-dimensional label indicator matrix is learned via spectral embedding, over which the k-means algorithm is run for acquiring the data labels. Furthermore, data features at different perspectives have different densities of semantic information, which contribute to the final clustering results with varying degrees, so we model this diversity via an adaptive weight learning scheme. Fig. 1 shows the overall framework of the proposed ULTLSE. Generally, the contributions of the present study are summarized as follows:

- We propose a unified framework for dealing with the multi-view subspace clustering problem, which integrates the low-rank tensor learning and spectral embedding. Thus, the dependence between the low-rank tensor and label indicator matrix can be considered and utilized.
- We preserve the principal components of the self-representation tensor through the t-SVD based tensor nuclear norm. At the same time, each view has a specific measurement, which is quantified as a weighting coefficient.
- The proposed ULTLSE is evaluated on eight real-world datasets with three categories. The experimental results indicate that ULTLSE outperforms other state-of-the-art single-view and multi-view clustering methods.

The remaining sections of this paper are structured as follows. In Section II, we briefly introduce related works relevant to our method. The notations and preliminaries are described in Section III. In Section IV, we elaborate upon the proposed ULTLSE for multi-view subspace clustering. Section V presents the experimental details. In Section VI, we summarize the paper and highlight the future research plans.

II. Related Work

A. Low-rank Analysis

For a matrix or a tensor, its low-rank parts contain the principal data information and correlation information of sample points. Since minimizing the rank of a matrix or tensor is a non-convex problem hard to resolve, researchers have found its optimal convex approximation, i.e., minimizing their nuclear norm. At present, low-rank analysis [25], [26] is widely researched and applied in data mining. For a few examples, Liu et al. [15] was committed to solving a low-rank representation matrix from the original feature matrix. Chang et al. [27] proposed an approximate low-rank factorized similarity learning approach incorporating affluent information from various sources in network. Tang et al. [28] utilized Tucker decomposition to complete the tri-clustered tensor for social image tag refinement. Tang et al. [29] designed a low-rank tensor learning method incorporating the anchor graph technology to achieve image retagging. Wu et al. [30] developed a low-rank kernelized hash functions optimization manner to tackle corrupted data. Fu et al. [31] recovered an essential representation tensor via low-rank approximation for multi-view clustering and semi-supervised classification.

B. Embedding Representation Learning

Embedding representation learning is an efficient technique for learning compact and discriminative data features, which is crucial for downstream tasks such as pattern recognition, vision processing, etc. Currently, there are two dominant models for embedding representation learning: shallow models and deep models. For example, Qi et al. [32] projected cross-modal data information onto a unified embedding space to capture the semantic correspondence. Hajjar et al. [33] explored a common nonnegative embedding with orthogonal constraints on the columns from multi-view data. Xie et al. [34] learned a low-dimensional embedding space from original data feature space, where the data were characterized by high discriminative capacity. Wang et al. [35] aimed to mine a unified subspace embedding representation from multiple features, and explored the complementary information using a diversity regularization. Cai et al. [36] enhanced the embedding discrimination via simultaneously considering contractive loss, clustering loss, and focal loss.

C. Multi-view Clustering

Multi-view clustering [37], [38], [39], [40] leverages the comprehensive information accumulated in multi-view data to cluster samples, and the acquisition of this information depends on effective feature fusion [41], [42], [43], [44]. To promote the clustering performance, researchers pay more and more attention to developing new feature fusion technologies and recovering the integrative data representation. Nowadays, there are many kinds of multi-view clustering approaches. Herein, we briefly review several major types of models.

Graph-based models are one of the most common models, which attempt to refine a uniformity affinity matrix of multi-view data. For instance, Nie et al. [45] learned the locality-preserving similarity matrix after graph fusion, which was equipped with c connected components. Zhan et al. [46] strengthened the consistency of various graphs via a disagreement cost function. Xu et al. [47] proposed a unified framework that simultaneously realized spectral embedding and nonnegative embedding, thus obtaining consistent clustering results. Tang et al. [48] utilized the diffusion process to encode the potential manifold geometry structure of samples. Wang et al. [49] learned a clustered representation via fusing varying views, and simultaneously encoded the local graph structure. Chen et al. [50] learned a uniformity graph matrix from low-rank subspace representation tensor with adaptive neighbor scheme.

Subspace learning aims to learn a low-dimensional embedding from original data, which reflects the principal information and is suitable for coping with high-dimensional data. Self-representation is the most commonly used model, whose mathematical form is written as

$$\mathbf{X} = \mathbf{X}\mathbf{Z} + \mathbf{E}, \tag{1}$$

where $\mathbf{X} \in \mathbb{R}^{d \times n}$ is the original data, $\mathbf{Z} \in \mathbb{R}^{n \times n}$ denotes the self-representation and $\mathbf{E} \in \mathbb{R}^{d \times n}$ is the error term. In multi-view cases, the objective is usually formulated as

$$\begin{aligned} &\min_{\mathbf{Z}^{(v)}, \mathbf{E}^{(v)}} \sum_{v=1}^{m} \Psi(\mathbf{Z}^{(v)}) + \lambda \sum_{v=1}^{m} \Omega(\mathbf{E}^{(v)}) \\ &\text{s.t. } \mathbf{X}^{(v)} = \mathbf{X}^{(v)}\mathbf{Z}^{(v)} + \mathbf{E}^{(v)}, v = 1, 2, \ldots, m, \end{aligned} \tag{2}$$

where $\Psi(\cdot)$ denotes the regularization operation, and $\Omega(\cdot)$ is a designed loss function. Diverse selections of regularization and loss function have formed multifarious multi-view subspace clustering methods. Wang et al. [51] enhanced the diversity of subspace representations via a position-aware exclusivity term. Zhang et al. [52] addressed the nonlinear structure problem in multi-view data through a robust low-rank kernel method. Tang et al. [53] endeavoured to explore a common low-rank affinity graph from multiple views, and learned a set of adaptive weights via diversity regularization. Moreover, the nonconvex low-rank tensor approximation manner was also studied [54], which was used to explore the informative components of the subspace representation tensor.

The semantic gap between varying data features may be very large, which is harmful to the feature fusion. Therefore, it is innovative to project different feature representations onto the same space by canonical correlation analysis (CCA). Chaudhuri et al. [55] learned a subspace representation from various views via CCA. Luo et al. [56] computed the covariance tensor to handle data with any number of views instead of only two views. Different from the above two works, Houthuys et al. [57] leveraged the kernel CCA to calculate the correlation errors.

III. Notations and Preliminaries

First of all, the meanings of some mathematical symbols are introduced in detail. Multi-view data is denoted by $\boldsymbol{\mathcal{X}} = \{\mathbf{X}^{(v)}\}_{v=1}^{m}$, and $\mathbf{X}^{(v)} \in \mathbb{R}^{d^{(v)} \times n}$. We use $\mathbf{a}$, $\mathbf{A}$, $\boldsymbol{\mathcal{A}}$ to denote a vector, a matrix, and a tensor, respectively. For a tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, $\boldsymbol{\mathcal{A}}_{ijk}$ denotes the (i, j, k)-th element and $\boldsymbol{\mathcal{A}}^{(i)}$ indicates the i-th frontal slice. $\boldsymbol{\mathcal{A}}^T \in \mathbb{R}^{n_2 \times n_1 \times n_3}$ is the transpose of $\boldsymbol{\mathcal{A}}$. $\hat{\boldsymbol{\mathcal{A}}} = fft(\boldsymbol{\mathcal{A}}, [\ ], 3)$ is the result of tensor $\boldsymbol{\mathcal{A}}$ after fast Fourier transformation (FFT) along the third dimension. Likewise, $\boldsymbol{\mathcal{A}}$ can be obtained by performing inverse FFT on $\hat{\boldsymbol{\mathcal{A}}}$, i.e., $\boldsymbol{\mathcal{A}} = ifft(\hat{\boldsymbol{\mathcal{A}}}, [\ ], 3)$. To understand the definition of t-SVD based tensor nuclear norm, several related definitions are necessary to introduce and listed below.

Definition 1: For a tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, the definitions of block diagonal matrix $bdiag(\boldsymbol{\mathcal{A}})$ and block circular matrix $bcirc(\boldsymbol{\mathcal{A}})$ are presented as follows

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

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

It can be seen that $\text{bdiag}(\boldsymbol{\mathcal{A}})$ is a new diagonal matrix by rearranging the frontal slices $\{\boldsymbol{\mathcal{A}}^{(v)}\}_{v=1}^{n_3}$ of tensor $\boldsymbol{\mathcal{A}}$ diagonally and $\text{bcirc}(\boldsymbol{\mathcal{A}})$ is formed by rearranging $\{\boldsymbol{\mathcal{A}}^{(v)}\}_{v=1}^{n_3}$ vertically n_3 times.

Definition 2: **(t-product)** Given two tensors $\boldsymbol{\mathcal{M}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ and $\boldsymbol{\mathcal{N}} \in \mathbb{R}^{n_2 \times n_4 \times n_3}$. Thus, the t-product $\boldsymbol{\mathcal{G}} \in \mathbb{R}^{n_1 \times n_4 \times n_3}$ of them is computed by

$$\boldsymbol{\mathcal{G}} = \boldsymbol{\mathcal{M}} * \boldsymbol{\mathcal{N}} = bvfold(bcirc(\boldsymbol{\mathcal{M}}) \cdot bvec(\boldsymbol{\mathcal{N}})), \tag{3}$$

where $bvec(\boldsymbol{\mathcal{N}}) = \left[\boldsymbol{\mathcal{N}}^{(1)}; \boldsymbol{\mathcal{N}}^{(2)}; \cdots; \boldsymbol{\mathcal{N}}^{(n_3)}\right] \in \mathbb{R}^{n_2 n_3 \times n_4}$ splices n_3 frontal slices of tensor $\boldsymbol{\mathcal{N}}$ vertically and $bvfold(bvec(\boldsymbol{\mathcal{G}})) = \boldsymbol{\mathcal{G}}$ reconstructs the matrix $bvec(\boldsymbol{\mathcal{G}})$ into a 3-order tensor.

Definition 3: **(Orthogonal tensor)** A tensor $\boldsymbol{\mathcal{P}} \in \mathbb{R}^{n_1 \times n_1 \times n_2}$ is orthogonal if it satisfies the following form

$$\boldsymbol{\mathcal{P}}^T * \boldsymbol{\mathcal{P}} = \boldsymbol{\mathcal{P}} * \boldsymbol{\mathcal{P}}^T = \boldsymbol{\mathcal{I}}, \tag{4}$$

where the tensor $\boldsymbol{\mathcal{I}} \in \mathbb{R}^{n_1 \times n_1 \times n_2}$ is termed as an identity tensor, whose first frontal slice is a $n_1 \times n_1$ identity matrix and the other frontal slices are all zeros.

Definition 4: **(t-SVD)** A tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ can be decomposed via t-SVD

$$\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{U}} * \boldsymbol{\mathcal{D}} * \boldsymbol{\mathcal{V}}^T, \tag{5}$$

where $\boldsymbol{\mathcal{U}} \in \mathbb{R}^{n_1 \times n_1 \times n_3}$ and $\boldsymbol{\mathcal{V}} \in \mathbb{R}^{n_2 \times n_2 \times n_3}$ are orthogonal, $\boldsymbol{\mathcal{D}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is f-diagonal. And f-diagonal tensor is a tensor whose each frontal slice is a diagonal matrix. Fig. 2 demonstrates the result of a 3-order tensor decomposed by t-SVD.

Definition 5: **(t-SVD based tensor nuclear norm** [25], [58]) The t-SVD based tensor nuclear norm of $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is defined as the sum of diagonal values of all frontal slices along with the minimum dimension in the f-diagonal $\hat{\boldsymbol{\mathcal{D}}}$.

$$\begin{aligned} ||\boldsymbol{\mathcal{A}}||_* &= ||bdiag(\hat{\boldsymbol{\mathcal{A}}})||_* = ||bdiag(\hat{\boldsymbol{\mathcal{D}}})||_* \\ &= \sum_{i=1}^{\min\{n_1, n_2\}} \sum_{j=1}^{n_3} |\hat{\boldsymbol{\mathcal{D}}}^{(j)}(i, i)|, \end{aligned} \tag{6}$$

where $\hat{\boldsymbol{\mathcal{D}}}^{(j)}$ can be solved via $\hat{\boldsymbol{\mathcal{A}}}^{(j)} = \hat{\boldsymbol{\mathcal{U}}}^{(j)} * \hat{\boldsymbol{\mathcal{D}}}^{(j)} * \hat{\boldsymbol{\mathcal{V}}}^{(j)^T}$.

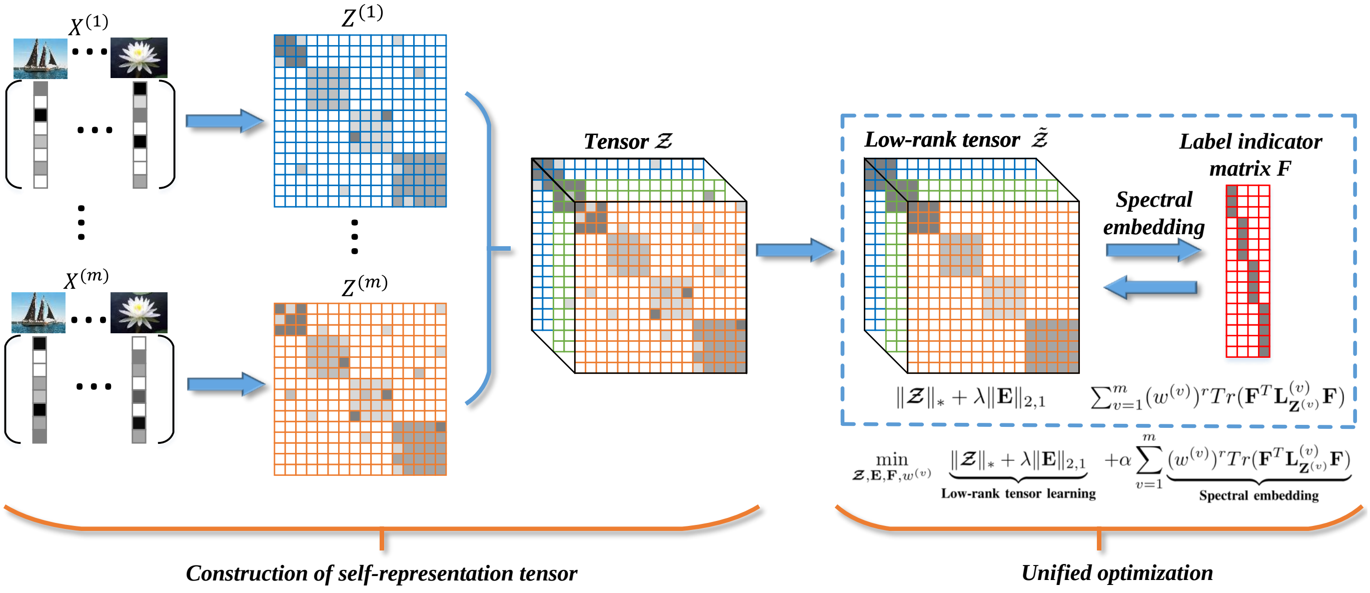

$X^{(1)}$
$Z^{(1)}$
$X^{(m)}$
$Z^{(m)}$
Tensor $\mathcal{Z}$
Low-rank tensor $\tilde{\mathcal{Z}}$
Label indicator matrix F
Spectral embedding
$\|\mathcal{Z}\|_* + \lambda\|\mathbf{E}\|_{2,1}$
$\sum_{v=1}^{m}(w^{(v)})^r Tr(\mathbf{F}^T\mathbf{L}^{(v)}_{\mathbf{Z}^{(v)}}\mathbf{F})$
$\min_{\mathcal{Z},\mathbf{E},\mathbf{F},w^{(v)}} \underbrace{\|\mathcal{Z}\|_* + \lambda\|\mathbf{E}\|_{2,1}}_{\text{Low-rank tensor learning}} + \alpha\sum_{v=1}^{m}\underbrace{(w^{(v)})^r Tr(\mathbf{F}^T\mathbf{L}^{(v)}_{\mathbf{Z}^{(v)}}\mathbf{F})}_{\text{Spectral embedding}}$
Construction of self-representation tensor
Unified optimization

Fig. 1: The overall framework of the proposed ULTLSE. Firstly, the subspace representation $\mathbf{Z}^{(v)}$ of each view is explored from the data features, and $\{\mathbf{Z}^{(v)}\}_{v=1}^{m}$ are aggregated into the tensor $\mathcal{Z}$ via *cat(·)* function. Thus, the low-rank tensor learning and spectral embedding are performed simultaneously, which is different from existing tensor based methods that separate the two processes without taking into account their inner dependence. After iteration, the label indicator matrix can be directly obtained.

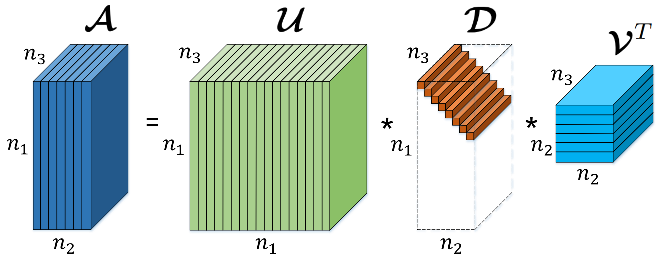


Fig. 2: The diagram of the t-SVD of a tensor with size $n_1 \times n_2 \times n_3$.

IV. Unified Low-rank Tensor Learning and Spectral Embedding for Multi-view Subspace Clustering

In this section, we first elaborate the proposed unified framework ULTLSE for dealing with multi-view subspace clustering problem. In addition, the optimization process based on the alternating direction method of multipliers (ADMM) is also explained.

A. Problem Formulation

The focus of multi-view learning is to mine the complementarity and consistency information from multi-view data. Most available multi-view methods find and utilize the properties from pairwise matrices by a local perspective. On the contrary, t-SVD-MSC [20] first constructs a tensor composed of self-representations of all views, then imposes the t-SVD based tensor nuclear norm on it. After low-rank tensor optimization, the high-order correlations are explored across all views. The objective of t-SVD-MSC is written as

$$\begin{aligned}
&\min_{\mathcal{Z},\mathbf{E}} \|\mathcal{Z}\|_* + \lambda\|\mathbf{E}\|_{2,1} \\
\text{s.t. } &\mathbf{X}^{(v)} = \mathbf{X}^{(v)}\mathbf{Z}^{(v)} + \mathbf{E}^{(v)}, v = 1, 2, \ldots, m, \\
&\mathcal{Z} = \Phi\left(\mathbf{Z}^{(1)}, \mathbf{Z}^{(2)}, \ldots, \mathbf{Z}^{(m)}\right), \\
&\mathbf{E} = \left[\mathbf{E}^{(1)}; \mathbf{E}^{(2)}; \ldots; \mathbf{E}^{(m)}\right],
\end{aligned} \tag{7}$$

where $\mathbf{E}$ is constructed via the vertical splicing of $\{\mathbf{E}^{(v)}\}_{v=1}^{m}$, and $\Phi(\cdot)$ represents the operation of combining multiple matrices with the same dimension into a tensor. $\|\cdot\|_{2,1}$ is the $l_{2,1}$-norm and defined as $\|\mathbf{X}\|_{2,1} = \sum_j \|\mathbf{X}(:,j)\|_2$, which makes the columns of $\mathbf{E}$ close to zero. After iterative optimization, the tensor $\tilde{\mathcal{Z}}$ with low-rank property can be obtained. Thus, the consensus affinity matrix is computed by $\frac{1}{m}\sum_{v=1}^{m}\left(|\mathbf{Z}^{(v)}| + |\mathbf{Z}^{(v)^T}|\right)/2$, then the low-dimensional label indicator matrix $\mathbf{F} \in \mathbb{R}^{n\times c}$ is further learned by spectral clustering, where c denotes the number of clusters.

From the above, we can observe that t-SVD-MSC yields the consensus affinity matrix by averaging the m frontal slices of $\tilde{\mathcal{Z}}$ and separates the solving processes of $\tilde{\mathcal{Z}}$ and $\mathbf{F}$. The practices have two obvious disadvantages. Firstly, each view depicts a certain aspect of objects, there must be differences between views. Therefore, averaging the tensor $\tilde{\mathcal{Z}}$ to gain the affinity matrix is unreasonable, which may affect the quality of the label indicator matrix. Moreover, exploring spectral embedding directly from multiple frontal slices $\{\mathbf{Z}^{(v)}\}_{v=1}^{m}$ in a low-rank tensor $\tilde{\mathcal{Z}}$ rather than the final affinity matrix is more conducive to leveraging the complementarity hidden in multi-view data. What needs to be emphasized that only one global spectral embedding matrix is learned. Based on such an idea, we propose the following method for the solution of the label indicator matrix $\mathbf{F} \in \mathbb{R}^{n\times c}$,

$$\begin{aligned}&\sum_{v=1}^{m}(w^{(v)})^{r}Tr(\mathbf{F}^{T}\mathbf{L}^{(v)}_{\mathbf{Z}^{(v)}}\mathbf{F})\\ &\text{s.t. } \mathbf{F}^{T}\mathbf{F}=\mathbf{I}, \mathbf{w}^{T}\mathbf{1}=1, w^{(v)}\geq 0, v=1,2,\ldots,m,\end{aligned} \tag{8}$$

where $\mathbf{w}$ is the weight vector composed of $\{w^{(v)}\}_{v=1}^{m}$, and r is a hyper parameter that adjusts the weight distribution. The Laplacian matrix $\mathbf{L}^{(v)}_{\mathbf{Z}^{(v)}}$ is defined as $\mathbf{D}^{(v)} - (\mathbf{Z}^{(v)} + \mathbf{Z}^{(v)^{T}})/2$, and $\mathbf{D}^{(v)}_{ii} = \sum_{j}(\mathbf{Z}^{(v)} + \mathbf{Z}^{(v)^{T}})_{ij}/2$. In Eq. (8), the unique spectral embedding $\mathbf{F}$ is learned from the set of $\{\mathbf{L}^{(v)}_{\mathbf{Z}^{(v)}}\}_{v=1}^{m}$, then the complementary information across multiple views can be injected into $\mathbf{F}$. Secondly, the solution of $\mathbf{F}$ depends on $\tilde{\boldsymbol{\mathcal{Z}}}$. If the two terms are solved separately, their correlation is fixed, it could not find the optimal label indicator matrix. However, if the value of $\mathbf{F}$ is adjusted dynamically with the optimization of $\tilde{\boldsymbol{\mathcal{Z}}}$ until convergence, it is more beneficial to get the optimal solution. Hence, we propose to integrate the low-rank tensor learning and spectral embedding as a unified framework. For enabling this goal, Eq. (7) and Eq. (8) are combined to construct the ultimate objective function:

$$\begin{aligned}&\min_{\boldsymbol{\mathcal{Z}},\mathbf{E},\mathbf{F},w^{(v)}} \underbrace{\|\boldsymbol{\mathcal{Z}}\|_{*} + \lambda\|\mathbf{E}\|_{2,1}}_{\textbf{Low-rank tensor learning}} + \alpha\sum_{v=1}^{m}\underbrace{(w^{(v)})^{r}Tr(\mathbf{F}^{T}\mathbf{L}^{(v)}_{\mathbf{Z}^{(v)}}\mathbf{F})}_{\textbf{Spectral embedding}}\\ &\text{s.t. } \mathbf{X}^{(v)} = \mathbf{X}^{(v)}\mathbf{Z}^{(v)} + \mathbf{E}^{(v)}, v=1,2,\ldots,m,\\ &\quad\mathbf{F}^{T}\mathbf{F}=\mathbf{I}, \mathbf{w}^{T}\mathbf{1}=1, w^{(v)}\geq 0,\\ &\quad\boldsymbol{\mathcal{Z}} = \Phi\left(\mathbf{Z}^{(1)},\mathbf{Z}^{(2)},\ldots,\mathbf{Z}^{(m)}\right),\\ &\quad\mathbf{E} = \left[\mathbf{E}^{(1)};\mathbf{E}^{(2)};\ldots;\mathbf{E}^{(m)}\right].\end{aligned} \tag{9}$$

Here, λ and α are two nonnegative parameters used to balance the three terms. In Eq. (9), the low-rank subspace representation tensor and low-dimensional label indicator matrix are learned at the meantime. After iteration, the latter can be obtained without any intermediate process, over which the k-means algorithm is applied for getting the clustering results.

B. Optimization Process

For solving Eq. (9), the ADMM method is introduced to obtain the optimal solution of each variable. Before optimization, we formulate the augmented Lagrangian function of Eq. (9) as:

$$\begin{aligned}&\boldsymbol{\mathcal{F}}\left(\{\mathbf{Z}^{(v)}\}_{v=1}^{m};\{\mathbf{E}^{(v)}\}_{v=1}^{m};\boldsymbol{\mathcal{H}};\{w^{(v)}\}_{v=1}^{m};\mathbf{F}\right)\\ &= \|\boldsymbol{\mathcal{H}}\|_{*} + \lambda\|\mathbf{E}\|_{2,1} + \alpha\sum_{v=1}^{m}(w^{(v)})^{r}Tr(\mathbf{F}^{T}\mathbf{L}^{(v)}_{\mathbf{Z}^{(v)}}\mathbf{F})\\ &+\sum_{v=1}^{m}\Big(\langle\mathbf{C}^{(v)},\mathbf{X}^{(v)}-\mathbf{X}^{(v)}\mathbf{Z}^{(v)}-\mathbf{E}^{(v)}\rangle\\ &+\frac{\theta}{2}||\mathbf{X}^{(v)}-\mathbf{X}^{(v)}\mathbf{Z}^{(v)}-\mathbf{E}^{(v)}\|_{F}^{2}\Big) + \langle\boldsymbol{\mathcal{T}},\boldsymbol{\mathcal{Z}}-\boldsymbol{\mathcal{H}}\rangle\\ &+\frac{\xi}{2}\|\boldsymbol{\mathcal{Z}}-\boldsymbol{\mathcal{H}}\|_{F}^{2},\end{aligned} \tag{10}$$

where $\boldsymbol{\mathcal{H}}$ is the adjunct variable, the matrices $\{\mathbf{C}^{(v)}\}_{v=1}^{m}$ and the tensor $\boldsymbol{\mathcal{T}}$ are Lagrange multipliers, θ and ξ denote two nonnegative parameters. Thus, we update various variables via the approaches presented below.

1) *Fix* $\mathbf{E}$, $\boldsymbol{\mathcal{H}}$, $w^{(v)}$, *and* $\mathbf{F}$ *to optimize* $\mathbf{Z}^{(v)}$: Focusing on the solution of one view, i.e., $\mathbf{Z}^{(v)}$, the problem becomes

$$\begin{aligned}&\min_{\mathbf{Z}^{(v)}}\alpha(w^{(v)})^{r}Tr(\mathbf{F}^{T}\mathbf{L}^{(v)}_{\mathbf{Z}^{(v)}}\mathbf{F})\\ &+\langle\mathbf{C}^{(v)},\mathbf{X}^{(v)}-\mathbf{X}^{(v)}\mathbf{Z}^{(v)}-\mathbf{E}^{(v)}\rangle\\ &+\frac{\theta}{2}||\mathbf{X}^{(v)}-\mathbf{X}^{(v)}\mathbf{Z}^{(v)}-\mathbf{E}^{(v)}\|_{F}^{2} + \langle\mathbf{T}^{(v)},\mathbf{Z}^{(v)}-\mathbf{H}^{(v)}\rangle\\ &+\frac{\xi}{2}\left\|\mathbf{Z}^{(v)}-\mathbf{H}^{(v)}\right\|_{F}^{2}.\end{aligned} \tag{11}$$

Setting the derivative of Eq. (11) with respect to $\mathbf{Z}^{(v)}$ to be zero, the solution of $\mathbf{Z}^{(v)}$ can be obtained by

$$\begin{aligned}\mathbf{Z}^{(v)^{*}} &= (\mu\mathbf{X}^{(v)^{T}}\mathbf{X}^{(v)}+\rho\mathbf{I})^{-1}(\alpha(w^{(v)})^{r}\mathbf{F}\mathbf{F}^{T}+\theta\mathbf{X}^{(v)^{T}}\mathbf{X}^{(v)}\\ &\quad-\theta\mathbf{X}^{(v)^{T}}\mathbf{E}^{(v)}+\mathbf{X}^{(v)^{T}}\mathbf{C}^{(v)}+\xi\mathbf{H}^{(v)}-\mathbf{T}^{(v)}).\end{aligned} \tag{12}$$

2) *Fix* $\{\mathbf{Z}^{(v)}\}_{v=1}^{m}$, $\boldsymbol{\mathcal{H}}$, $\{w^{(v)}\}_{v=1}^{m}$, *and* $\mathbf{F}$ *to optimize* $\mathbf{E}$: The problem becomes

$$\begin{aligned}&\min_{\mathbf{E}}\lambda\|\mathbf{E}\|_{2,1}+\sum_{v=1}^{m}\Big(\langle\mathbf{C}^{(v)},\mathbf{X}^{(v)}-\mathbf{X}^{(v)}\mathbf{Z}^{(v)}-\mathbf{E}^{(v)}\rangle\\ &+\frac{\theta}{2}||\mathbf{X}^{(v)}-\mathbf{X}^{(v)}\mathbf{Z}^{(v)}-\mathbf{E}^{(v)}\|_{F}^{2}\Big)\\ &=\min_{\mathbf{E}}\lambda||\mathbf{E}||_{2,1}+\sum_{v=1}^{m}\frac{\theta}{2}||\mathbf{E}-(\mathbf{X}^{(v)}-\mathbf{X}^{(v)}\mathbf{Z}^{(v)}+\frac{1}{\theta}\mathbf{C}^{(v)})||_{F}^{2}\\ &=\min_{\mathbf{E}}\lambda||\mathbf{E}||_{2,1}+\frac{\theta}{2}||\mathbf{E}-\mathbf{P}||_{F}^{2},\end{aligned} \tag{13}$$

where $\mathbf{P} = [\mathbf{P}^{(1)};\mathbf{P}^{(2)};\cdots;\mathbf{P}^{(m)}]$ and $\mathbf{P}^{(v)} = \mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}^{(v)} + \frac{1}{\theta}\mathbf{C}^{(v)}$. Following the solution proposed in the literature [15], $\mathbf{E}$ can be updated via

$$\mathbf{E}^{*}_{:,j} = \begin{cases}\frac{||\mathbf{P}_{:,j}||_{2}-\frac{\lambda}{\theta}}{||\mathbf{P}_{:,j}||_{2}}\mathbf{P}_{:,j}, & \left\|\mathbf{P}_{:,j}\right\|_{2} > \frac{\lambda}{\theta}\\ \mathbf{0}, & \text{otherwise.}\end{cases} \tag{14}$$

3) *Fix* $\{\mathbf{Z}^{(v)}\}_{v=1}^{m}$, $\mathbf{E}$, $\{w^{(v)}\}_{v=1}^{m}$, *and* $\mathbf{F}$ *to optimize* $\boldsymbol{\mathcal{H}}$: While keeping the related terms, the optimization of $\boldsymbol{\mathcal{H}}$ is to address the problem

$$\begin{aligned}&\min_{\boldsymbol{\mathcal{H}}}||\boldsymbol{\mathcal{H}}||_{*}+\langle\boldsymbol{\mathcal{T}},\boldsymbol{\mathcal{Z}}-\boldsymbol{\mathcal{H}}\rangle+\frac{\xi}{2}\|\boldsymbol{\mathcal{Z}}-\boldsymbol{\mathcal{H}}\|_{F}^{2}\\ &=\min_{\boldsymbol{\mathcal{H}}}||\boldsymbol{\mathcal{H}}||_{*}+\frac{\xi}{2}||\boldsymbol{\mathcal{H}}-(\boldsymbol{\mathcal{Z}}+\frac{1}{\xi}\boldsymbol{\mathcal{T}})||_{F}^{2}.\end{aligned} \tag{15}$$

For optimizing $\boldsymbol{\mathcal{H}}$ more effectively, we rotate the size of $\boldsymbol{\mathcal{H}}$ from $n\times n\times m$ to $n\times m\times n$. The rotation operation is necessary. According to Eq. (6) above, SVD is performed on a tensor's each frontal slice after FFT when computing the tensor nuclear norm. After rotation, as shown in Fig. 3, the cross-view low-rank properties, that is, the global correlation can be preserved. Inspired by [59], the close-form solution of Eq. (15) is derived via

$$\boldsymbol{\mathcal{H}}^{*} = \boldsymbol{\mathcal{U}} * \boldsymbol{\mathcal{C}}_{m/\xi}(\boldsymbol{\mathcal{D}}) * \boldsymbol{\mathcal{V}}^{T}, \tag{16}$$

where $\mathcal{C}_{m/\xi}(\mathcal{D}) = \mathcal{D} * \mathcal{Q}$. $\mathcal{Q}$ is an f-diagonal tensor and its diagonal entry in the Fourier domain is $\hat{\mathcal{Q}}(i,i,k) = max\{1 - \frac{m/\xi}{\mathcal{D}(i,i,k)}, 0\}$.

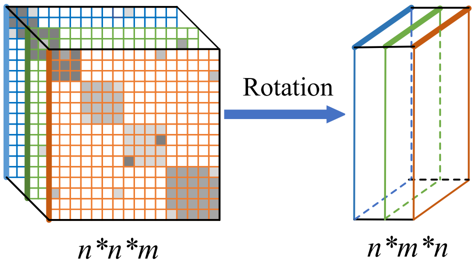


Fig. 3: The diagram of tensor rotation. The tensor size before rotation is $n*n*m$, while the size after rotation is transformed to $n*m*n$.

4) *Fix* $\{\mathbf{Z}^{(v)}\}_{v=1}^{m}$, $\mathcal{H}$, $\mathbf{E}$, *and* $\mathbf{F}$ *to optimize* $\{w^{(v)}\}_{v=1}^{m}$: The problem is transformed into the form

$$\begin{aligned} &\min_{w^{(v)}} \sum_{v=1}^{m} (w^{(v)})^r Tr(\mathbf{F}^T \mathbf{L}_{\mathbf{Z}^{(v)}}^{(v)} \mathbf{F}) \\ &\text{s.t. } \mathbf{w}^T \mathbf{1} = 1, w^{(v)} \geq 0. \end{aligned} \tag{17}$$

Denoting $\mathbf{B}^{(v)} = Tr(\mathbf{F}^T \mathbf{L}_{\mathbf{Z}^{(v)}}^{(v)} \mathbf{F})$, we obtain the Lagrangian function of Eq. (17) as follows:

$$\mathcal{L} = \sum_{v=1}^{m} (w^{(v)})^r \mathbf{B}^{(v)} - \eta(\sum_{v=1}^{m} w^{(v)} - 1). \tag{18}$$

Deriving the derivative of $w^{(v)}$ in Eq. (18) and setting the derivative value to zero, we have

$$w^{(v)^*} = \frac{(\mathbf{B}^{(v)})^{1/(1-r)}}{\sum_{v=1}^{m} (\mathbf{B}^{(v)})^{1/(1-r)}}. \tag{19}$$

5) *Fix* $\{\mathbf{Z}^{(v)}\}_{v=1}^{m}$, $\mathbf{E}$, $\mathcal{H}$, *and* $\{w^{(v)}\}_{v=1}^{m}$ *to optimize* $\mathbf{F}$*:* We have the following problem

$$\begin{aligned} &\min_{\mathbf{F}} Tr(\mathbf{F}^T \mathbf{L} \mathbf{F}) \\ &\text{s.t. } \mathbf{F}^T \mathbf{F} = \mathbf{I}, \end{aligned} \tag{20}$$

where $\mathbf{L} = \sum_{v=1}^{m} (w^{(v)})^r \mathbf{L}_{\mathbf{Z}^{(v)}}^{(v)}$. Thus, the eigenvectors corresponding to the first c smallest eigenvalues of $\mathbf{L}$ constitute the updated $\mathbf{F}^*$. To ensure that $\mathbf{L}_{\mathbf{Z}^{(v)}}^{(v)}$ is positive semi-definite, we perform a truncation strategy for $\mathbf{Z}^{(v)}$, that is, negative elements in $\mathbf{Z}^{(v)}$ are set to 0.

6) *Update the Lagrange multipliers* $\mathbf{C}^{(v)}$, $\mathcal{T}$ *and nonnegative parameters* θ, ξ,

$$\begin{aligned} \mathbf{C}^{(v)^*} &= \mathbf{C}^{(v)} + \mu(\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}^{(v)} - \mathbf{E}^{(v)}), \\ \mathcal{T}^* &= \mathcal{T} + \rho(\mathcal{Z} - \mathcal{H}), \\ \theta^* &= min(\psi * \theta, \theta_{max}), \\ \xi^* &= min(\psi * \xi, \xi_{max}), \end{aligned} \tag{21}$$

where ψ is a constant to adjust the convergence speed, θ_{max} and ξ_{max} represent the maximal values of θ and ξ. The main algorithm steps are summarized in Algorithm 1.

Algorithm 1 Unified Low-rank Tensor Learning and Spectral Embedding for Multi-view Subspace Clustering (ULTLSE)

```
Input: Multi-view data $\mathcal{X} = \{\mathbf{X}^{(v)}\}_{v=1}^{m}$, $\mathbf{X}^{(v)} \in \mathbb{R}^{d^{(v)} \times n}$,
    regularization parameters $\lambda$, $\alpha$.
Initialize $\mathcal{Z}_0 = \mathcal{H}_0 = \mathcal{T}_0 = \mathbf{0}$, $\mathbf{E}_0 = \mathbf{0}$, $\mathbf{C}_0^{(v)} = \mathbf{0}$,
    $w_0^{(v)} = \frac{1}{m}$, $\psi = 2$, $\varepsilon = 10^{-7}$, $\theta_0$, $\xi_0$, $\theta_{max} = \xi_{max} = 10^{10}$, $t = 0$.
while not convergent do
    for $v$ = 1 to $m$ do
        Update $\mathbf{Z}_{t+1}^{(v)}$ by Eq. (12);
    end for
    Update $\mathbf{E}_{t+1}$ by Eq. (14);
    Update $\mathcal{H}_{t+1}$ by Eq. (16);
    Update $w_{t+1}^{(v)}$ by Eq. (19);
    Update $\mathbf{F}_{t+1}$ by Eq. (20);
    Update $\mathbf{C}_{t+1}^{(v)}$, $\mathcal{T}_{t+1}$, $\theta_{t+1}$ and $\xi_{t+1}$ by Eq. (21);
    Check the convergence conditions:
    $||\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}_{t+1}^{(v)} - \mathbf{E}_{t+1}^{(v)}||_\infty \leq \varepsilon$,
    $||\mathbf{Z}_{t+1}^{(v)} - \mathbf{H}_{t+1}^{(v)}||_\infty \leq \varepsilon$,
    $||\mathbf{F}_{t+1} - \mathbf{F}_t||_\infty \leq \varepsilon$.
    ▷ When all three losses satisfy the above conditions,
    the model reaches the convergence, then the loop ends,
    otherwise it continues;
    $t = t + 1$;
end while
Output: Label indicator matrix $\mathbf{F}$.
▷ Perform k-means algorithm over $\mathbf{F}$ for obtaining the
    clustering results.
```

V. Experiments

A. Datasets Descriptions

In experiments, we collect eight datasets that can be divided into three categories: text, scene, and object. Specifically, text type includes 3Sources, BBCnews, and WikipediaArticles, scene type includes Scene-15 and MITIndoor-67, object type includes ALOI, Caltech-20, and Caltech-101. The details of these datasets are described below, and Table I also presents the detailed information.

3Sources [1] is comprised of 169 reports published at three prestigious news websites: Reuters, BBC, and Guardian, which cover six fields of technology, business, health, entertainment, politics, and sport.

BBCnews [2] contains 685 news reports with five themes, including sport, politics, business, entertainment, and technology. Each item is represented by four types of features.

WikipediaArticles [3] is an article dataset that consists of 693 documents with 10 classes, and each entry has two feature representations.

Scene-15 [4] is composed of 4,485 indoor and outdoor scene images spanning 15 classes. Three types of image features

[1] http://mlg.ucd.ie/datasets/3sources.html

[2] http://mlg.ucd.ie/datasets/bbc.html

[3] http://lig-membres.imag.fr/grimal/data.html

[4] http://www-cvr.ai.uiuc.edu/ponce_grp/data/

are extracted from each image, including PRI-CoLBP, CENTRIST, and PHOW.

MITIndoor-67 is an indoor image collection of 5,360 images with 67 categories. Except for the same three features as Scene-15, it has a peculiar feature representation extracted by VGG-VD network.

ALOI [5] consists of 1,079 images of 10 objects, which are photographed from various rotation angles and lighting conditions. Four different views include HSV color histograms, Haralick texture features, RGB color histograms and color similarities.

TABLE I: Statistics of eight datasets.

Datasets	Samples	Views	Clusters	Category
3Sources	169	3	6	Text
BBCnews	685	4	5	Text
WikipediaArticles	693	2	10	Text
Scene-15	4,485	3	15	Scene
MITIndoor-67	5,360	4	67	Scene
ALOI	1,079	4	10	Object
Caltech-20	2,386	6	20	Object
Caltech-101	8,677	4	101	Object

Caltech [6] is a collection with a large number of object images. In the experiments, we adopt two versions of the dataset. Specifically, **Caltech-20** is composed of 2,386 images of 20 categories from six views: wavelet moments (WM), GIST, CENTRIST, HOG, Gabor, and LBP features. **Caltech-101** contains 8,677 images with 101 kinds of objects from four different features: PHOW, LBP, CENTRIST and the feature representation extracted via Inception V3 network.

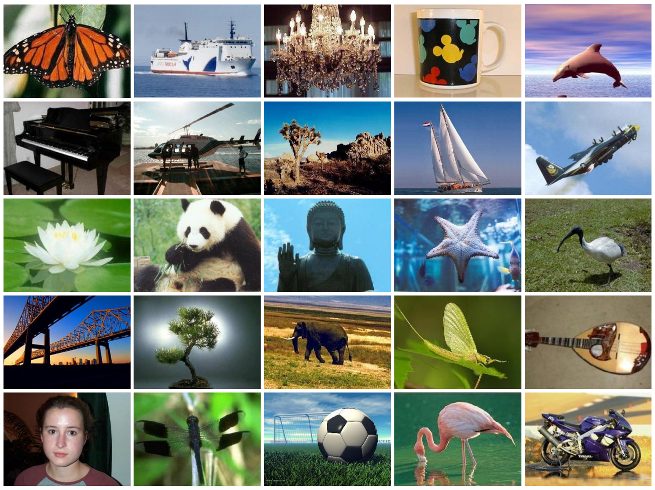


Fig. 4: Sample images from the dataset Caltech, which contains 101 different objects such as butterfly, steamship, chandelier, cup, dolphin.

B. Compared Algorithms and Parameter Settings

In order to make the experiments more objective and comprehensive, we choose nine state-of-the-art multi-view clustering approaches with three categories: graph-based methods (AMGL, MLAN, and MCGC), subspace-based approaches (ECMSC, CSMSC, MSC-IAS, and MCLES), and tensor-based approaches (LTMSC and t-SVD-MSC). Besides, we also compare the proposed model with traditional single-view methods such as SPC and LRR.

SPC$_{\text{best}}$ utilizes the most informative view for clustering and obtains the best performance via spectral clustering.

LRR$_{\text{best}}$ [15] achieves the optimal clustering performance through the low-rank subspace representation method with the highest quality view.

AMGL [60] extends the objective function of standard spectral clustering to multi-view cases, and explores the unique spectral embedding matrix in the new objective function.

MLAN [45] learns a global similarity matrix incorporating local structure information of data, which aims to alleviate the influence of noise and outlier samples.

MCGC [46] minimizes discrepancy between varying views to obtain a consistent graph, whose rank of Laplacian matrix is constrained with the value same as the cluster number.

ECMSC [51] mines the information complementarity in multiple views by a position-aware exclusivity term, and the label indicator matrix is solved with a consistency term.

CSMSC [16] treats multi-view subspace representations as the combination of a batch of specific representations and a consistent representation.

MSC-IAS [61] learns an intactness-aware similarity matrix in an intact space through the HSIC regularization.

MCLES [13] exploits a latent embedding space from multiple features, based on which the similarity matrix and cluster indicator matrix are learned at the same time.

LTMSC [19] constructs a 3-order tensor that is comprised of the self-representations of multiple views, then the rank of the tensor is minimized by SNN to preserve the critical components.

t-SVD-MSC [20] combines all views' self-representations into a 3-order tensor, whose low-rank components are encoded by the t-SVD based tensor nuclear norm.

Additionally, some parameter settings in above compared methods are described here. In MLAN, each sample is assigned to 9 nearest neighbors and the parameter λ ranges in [1, 20]. In MCGC, we tune the parameter β in [10, 100]. In ECMSC, the parameters α, β range [0.1, 0.5], [0.2, 0.7], respectively, and η is set 1.2. In CSMSC, we adjust the parameters λ_C and λ_D by varying in [0.001, 1]. In MSC-IAS, the dimension of latent intact space data is set 500, the nearest neighbor number k ranges in [3, 10], and the parameter λ_2 is tuned in [0.05, 1]. In MCLES, the parameters α, β, γ, and d vary in [0.2, 2], [0.2, 2], [0.001, 0.01], and [10, 100], respectively. In LTMSC, we tune the parameter γ in [0.1, 10]. In t-SVD-MSC, we vary the parameter λ in [0.01, 2].

C. Evaluation Metrics

To quantify the clustering results, clustering accuracy (ACC), normalized mutual information (NMI), adjusted rand index (ARI), F-score, precision, and recall are used to evaluate the performance of algorithms. In particular, the larger values demonstrate the better performance for the above metrics. Their calculation rules are described below.

[5] https://elki-project.github.io/datasets/multi-view

[6] http://www.vision.caltech.edu/Image_ Datasets/Caltech101

TABLE II: Performance of various clustering methods on text datasets 3Sources, BBCnews, and WikipediaArticles.

Datasets	Methods	ACC	NMI	ARI	F-score	Precision	Recall
3Sources	SPC$_{\rm best}$	0.572±0.024	0.459±0.022	0.349±0.030	0.491±0.021	0.525±0.035	0.461±0.011
	LRR$_{\rm best}$	0.610±0.026	0.462±0.003	0.406±0.023	0.568±0.019	0.439±0.015	0.670±0.043
	AMGL	0.341±0.037	0.072±0.031	-0.017±0.019	0.348±0.007	0.226±0.008	0.774±0.079
	MLAN	0.763±0.000	0.656±0.000	0.571±0.000	0.683±0.000	0.609±0.000	0.777±0.000
	MCGC	0.296±0.000	0.079±0.000	-0.040±0.000	0.335±0.000	0.216±0.000	0.744±0.000
	ECMSC	0.346±0.025	0.132±0.006	0.011±0.031	0.295±0.013	0.240±0.019	0.391±0.043
	CSMSC	0.630±0.003	0.443±0.007	0.502±0.005	0.627±0.003	0.580±0.006	0.683±0.002
	MSC-IAS	0.623±0.003	0.567±0.009	0.429±0.016	0.546±0.012	0.621±0.016	0.487±0.009
	MCLES	0.640±0.042	0.437±0.058	0.366±0.070	0.558±0.041	0.433±0.050	0.790±0.020
	LTMSC	0.782±0.003	0.699±0.008	0.658±0.008	0.739±0.006	**0.726±0.011**	0.725±0.007
	t-SVD-MSC	0.781±0.000	0.678±0.000	0.658±0.000	0.745±0.000	0.683±0.000	0.818±0.000
	ULTLSE	**0.817±0.005**	**0.711±0.010**	**0.717±0.004**	**0.789±0.009**	0.720±0.006	**0.873±0.013**
BBCnews	SPC$_{\rm best}$	0.438±0.002	0.295±0.001	0.204±0.001	0.399±0.000	0.382±0.002	0.417±0.003
	LRR$_{\rm best}$	0.802±0.000	0.568±0.000	0.621±0.000	0.712±0.000	0.697±0.000	0.727±0.000
	AMGL	0.344±0.026	0.016±0.009	0.004±0.011	0.373±0.004	0.236±0.004	0.893±0.099
	MLAN	0.853±0.007	0.698±0.010	0.716±0.005	0.783±0.004	0.776±0.003	0.790±0.004
	MCGC	0.350±0.000	0.039±0.000	0.001±0.000	0.373±0.000	0.235±0.000	0.903±0.000
	ECMSC	0.308±0.028	0.047±0.009	0.008±0.018	0.322±0.017	0.239±0.009	0.497±0.064
	CSMSC	0.917±0.000	0.770±0.000	0.807±0.000	0.853±0.000	0.847±0.000	0.859±0.000
	MSC-IAS	0.820±0.001	0.632±0.001	0.647±0.002	0.728±0.001	0.741±0.001	0.715±0.002
	MCLES	0.706±0.012	0.482±0.017	0.706±0.012	0.474±0.032	0.626±0.020	0.508±0.024
	LTMSC	0.579±0.000	0.424±0.006	0.401±0.003	0.547±0.003	0.524±0.002	0.572±0.004
	t-SVD-MSC	0.958±0.000	0.866±0.000	0.900±0.000	0.923±0.000	0.925±0.000	0.921±0.000
	ULTLSE	**0.994±0.000**	**0.977±0.000**	**0.985±0.000**	**0.989±0.000**	**0.990±0.000**	**0.987±0.000**
WikipediaArticles	SPC$_{\rm best}$	0.552±0.001	0.519±0.000	0.410±0.000	0.473±0.000	0.485±0.000	0.462±0.000
	LRR$_{\rm best}$	0.554±0.000	0.521±0.000	0.417±0.000	0.479±0.000	0.491±0.000	0.468±0.000
	AMGL	0.531±0.037	0.494±0.019	0.335±0.027	0.417±0.021	0.371±0.032	0.480±0.029
	MLAN	0.182±0.000	0.059±0.000	0.005±0.000	0.154±0.000	0.112±0.000	0.244±0.000
	MCGC	0.502±0.000	0.418±0.000	0.265±0.000	0.362±0.000	0.299±0.000	0.456±0.000
	ECMSC	0.561±0.000	0.516±0.000	0.411±0.000	0.472±0.000	0.493±0.000	0.454±0.000
	CSMSC	0.474±0.009	0.356±0.008	0.290±0.011	0.364±0.010	0.381±0.010	0.359±0.010
	MSC-IAS	0.463±0.014	0.428±0.013	0.294±0.017	0.372±0.015	0.368±0.017	0.377±0.016
	MCLES	0.543±0.003	0.474±0.004	0.359±0.005	0.430±0.004	0.421±0.005	0.440±0.004
	LTMSC	0.531±0.003	0.495±0.005	0.407±0.002	0.471±0.002	0.481±0.002	0.461±0.003
	t-SVD-MSC	0.513±0.002	0.475±0.004	0.386±0.003	0.452±0.003	0.439±0.003	0.464±0.002
	ULTLSE	**0.574±0.001**	**0.531±0.002**	**0.430±0.001**	**0.492±0.001**	**0.491±0.001**	**0.494±0.001**

TABLE III: Performance of various clustering methods on scene datasets Scene-15 and MITIndoor-67.

Datasets	Methods	ACC	NMI	ARI	F-score	Precision	Recall
Scene-15	SPC$_{\rm best}$	0.437±0.015	0.421±0.010	0.270±0.010	0.321±0.022	0.314±0.016	0.329±0.020
	LRR$_{\rm best}$	0.445±0.013	0.426±0.018	0.272±0.015	0.324±0.010	0.316±0.015	0.333±0.015
	AMGL	0.402±0.040	0.455±0.038	0.263±0.058	0.340±0.048	0.228±0.041	0.695±0.057
	MLAN	0.332±0.000	0.475±0.000	0.151±0.000	0.248±0.000	0.150±0.000	0.731±0.000
	MCGC	0.284±0.000	0.325±0.000	0.160±0.000	0.258±0.000	0.153±0.000	0.817±0.000
	ECMSC	0.457±0.001	0.463±0.002	0.303±0.001	0.357±0.001	0.318±0.001	0.408±0.001
	CSMSC	0.495±0.007	0.532±0.004	0.367±0.005	0.415±0.005	0.377±0.003	0.462±0.008
	MSC-IAS	0.583±0.003	0.603±0.003	0.429±0.006	0.472±0.006	0.438±0.009	0.512±0.013
	MCLES	-	-	-	-	-	-
	LTMSC	0.574±0.009	0.571±0.011	0.424±0.010	0.465±0.007	0.452±0.003	0.479±0.008
	t-SVD-MSC	0.812±0.007	0.858±0.007	0.771±0.003	0.788±0.001	0.743±0.006	0.839±0.003
	ULTLSE	**0.868±0.004**	**0.894±0.006**	**0.844±0.020**	**0.855±0.009**	**0.854±0.003**	**0.855±0.005**
MITIndoor-67	SPC$_{\rm best}$	0.443±0.011	0.559±0.009	0.304±0.011	0.315±0.013	0.294±0.010	0.340±0.014
	LRR$_{\rm best}$	0.120±0.004	0.226±0.006	0.031±0.007	0.045±0.004	0.044±0.006	0.047±0.004
	AMGL	0.146±0.009	0.232±0.015	0.037±0.004	0.063±0.004	0.035±0.002	0.340±0.019
	MLAN	0.468±0.010	0.611±0.003	0.312±0.006	0.323±0.006	0.299±0.008	0.352±0.003
	MCGC	0.081±0.000	0.118±0.000	0.009±0.000	0.038±0.000	0.020±0.000	0.581±0.000
	ECMSC	0.353±0.002	0.489±0.001	0.216±0.002	0.228±0.001	0.213±0.001	0.247±0.002
	CSMSC	0.401±0.012	0.513±0.005	0.250±0.004	0.262±0.004	0.247±0.006	0.280±0.002
	MSC-IAS	0.333±0.006	0.466±0.002	0.176±0.004	0.189±0.004	0.174±0.004	0.207±0.004
	MCLES	-	-	-	-	-	-
	LTMSC	0.431±0.002	0.546±0.004	0.280±0.008	0.290±0.002	0.279±0.006	0.306±0.005
	t-SVD-MSC	0.684±0.005	0.750±0.007	0.555±0.005	0.562±0.008	0.543±0.005	0.582±0.004
	ULTLSE	**0.795±0.023**	**0.911±0.017**	**0.755±0.020**	**0.759±0.021**	**0.688±0.022**	**0.844±0.012**

Let τ_i and p_i denote the truth label and prediction label of the i-th sample, respectively. n represents the number of data points. Then ACC is defined as

$$\mathrm{ACC}=\frac{\sum_{i=1}^{n}\delta\left(\tau_i,\operatorname{map}\left(p_i\right)\right)}{n}, \tag{22}$$

where map$(\cdot)$ denotes the optimal mapping function that permutes the prediction labels to match the truth labels, $\delta(\cdot,\cdot)$ is a discriminant function and has following property

$$\delta(a,b)=\begin{cases}1, & \text{if } a=b;\\ 0, & \text{otherwise.}\end{cases}$$

NMI presents the indication of the shared mutual information between two clusters and can be calculated by the following confusion matrix

$$\mathrm{NMI}(\mathbf{Q},\mathbf{P})=\frac{\sum_{i=1}^{\tilde{c}}\sum_{j=1}^{c}\left|P_i\cap Q_j\right|\log\frac{n\left|P_i\cap Q_j\right|}{\left|P_i\right|\left|Q_j\right|}}{\sqrt{\left(\sum_{i=1}^{\tilde{c}}\left|P_i\right|\log\frac{\left|P_i\right|}{n}\right)\left(\sum_{j=1}^{c}\left|Q_j\right|\log\frac{\left|Q_j\right|}{n}\right)}}, \tag{23}$$

where $\mathbf{Q}=\{Q_j\}_{j=1}^{c}$, $\mathbf{P}=\{P_i\}_{i=1}^{\tilde{c}}$ denote the set of truth label and the set of prediction label, respectively.

ARI is a distance metric to measure the similarity degree between two clusters. It is defined as

$$\mathrm{ARI}=\frac{A-\frac{BC}{n(n-1)/2}}{(1/2)(B+C)-\frac{BC}{n(n-1)/2}}, \tag{24}$$

where $A=\sum_{i,j}\frac{K_{ij}(K_{ij}-1)}{2}$, $B=\sum_{i}\frac{K_i(K_i-1)}{2}$ and $C=\sum_{j}\frac{K_j(K_j-1)}{2}$. Specifically, K_{ij} represents the number of samples that should be partitioned into the i-th cluster but partitioned into the j-th cluster. K_i, K_j denote the quantity of data points belonging to the i-th and j-th cluster, respectively.

Let TP be the number of correctly labeled positive cluster samples and FP be wrongly labeled positive cluster samples. Thus, the calculation formula of precision is written as

$$\text{Precision }=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}. \tag{25}$$

Furthermore, by introducing the variable FN that refers to the number of entries wrongly labeled negative cluster, recall is computed by

$$\text{Recall}=\frac{\mathrm{TP}}{\mathrm{FP}+\mathrm{FN}}. \tag{26}$$

F-score is defined by the harmonic mean of precision and recall

$$\text{F-score}=2\frac{\text{Precision}\cdot\text{Recall}}{\text{Precision }+\text{ Recall}}. \tag{27}$$

D. Experimental Results and Discussions

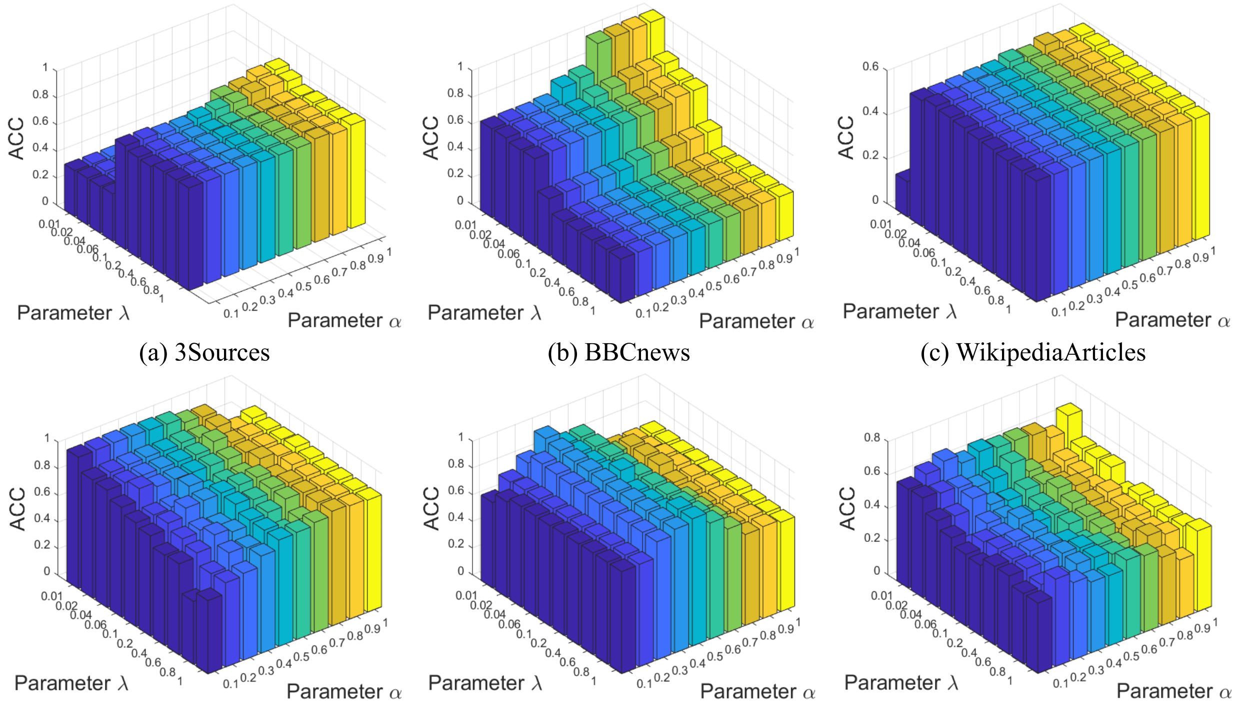
1) Performance overview. We list the detailed experimental results in Tables II-IV, where bold values indicate the best results. Since the running time of MCLES on the datasets Scene-15, MITIndoor-67, and Caltech-101 is too long, we use the symbol ”-” to replace the results. From the numerical values, we can obtain some insights:

- The proposed ULTLSE outperforms other clustering approaches in most cases. In particular, compared with the second-best method t-SVD-MSC, ULTLSE improves by 11.1%, 16.1%, 20.0%, 19.7%, 14.5%, 26.2% in terms of six metrics on the dataset MITIndoor, respectively. Similarly, ULTLSE achieves 7.2%, 10.0%, 10.8%, 9.7%, 20.5% improvements with respect to the five metrics on the dataset ALOI except for Precision. At the same time, our model performs better than LTMSC on all datasets, which also belongs to tensor-based methods. t-SVD-MSC and LTMSC separate the learning process of the low-rank tensor from that of the label indicator matrix, thereby ignoring the dependency between them. This practice makes the learned label indicator matrix less accurate and directly affects the final clustering results. Furthermore, they treat all views indiscriminately, which can not enhance the contributions of high-quality views and makes the results susceptible to interference from low-quality views. In view of the above disadvantages, ULTLSE integrates low-rank tensor learning and spectral embedding into a joint model, where the label indicator matrix can be directly obtained after iteration. Also, ULTLSE learns an adaptive weight for each view to distinguish the contributions of different feature representations.
- Multi-view clustering methods are capable of segmenting data points better than single-view methods. This is because single-view data lacks a more comprehensive description of objects while multi-view data has richer semantic information, which is explored and utilized by multi-view approaches. However, it can also be seen that many multi-view methods perform worse than single-view methods on the dataset WikipediaArticles. The main reason may be that the differences between views are large and it is difficult to find effective consistent information.
- The tensor-based method LTMSC does not perform as well as t-SVD-MSC and ULTLSE on most datasets, which uses SNN to constrain the rank of the target tensor while the latter two methods adopt t-SVD based nuclear norm. SNN is a loose substitute of Tucker rank and has difficulty to uncovering the global structure. However, the t-SVD based nuclear norm implements SVD decomposition to the entire tensor, then the complementary and consistent information can be explored across all views.
- The graph-based methods AMGL and MCGC do not perform well on the datasets 3Sources, Scene-15, MITIndoor-67, and Caltech-101, which have higher dimensions or possess more samples than other datasets. Datasets with these characteristics contain more redundant information and noise that could have a great impact on the construction of graphs.

2) The necessity of weights $\{w^{(v)}\}_{v=1}^{m}$. As shown in Table V, we can see that different feature representations produce varying performance even for the same dataset via spectral clustering. The differences between various views in the same dataset could also be very large. Furthermore, taking the dataset WikipediaArticles as an example, the weights of two views are 0.3024 and 0.6976, respectively, which matches the performance of each view. Therefore, it is necessary to

TABLE IV: Performance of various clustering methods on object datasets ALOI, Caltech-20, and Caltech-101.

Datasets	Methods	ACC	NMI	ARI	F-score	Precision	Recall
ALOI	SPC_{best}	0.580±0.020	0.718±0.015	0.535±0.019	0.592±0.015	0.479±0.025	0.777±0.012
	LRR_{best}	0.602±0.021	0.554±0.007	0.458±0.009	0.515±0.008	0.490±0.010	0.542±0.007
	AMGL	0.522±0.040	0.549±0.017	0.332±0.024	0.410±0.019	0.347±0.026	0.503±0.010
	MLAN	0.547±0.016	0.551±0.026	0.302±0.045	0.404±0.012	0.277±0.014	0.768±0.024
	MCGC	0.657±0.000	0.626±0.000	0.421±0.000	0.499±0.000	0.367±0.000	0.779±0.000
	ECMSC	0.645±0.000	0.611±0.000	0.423±0.000	0.491±0.000	0.413±0.000	0.606±0.000
	CSMSC	0.756±0.000	0.733±0.000	0.636±0.000	0.674±0.000	0.638±0.000	0.714±0.000
	MSC-IAS	0.613±0.039	0.648±0.015	0.479±0.027	0.536±0.023	0.487±0.032	0.597±0.025
	MCLES	0.497±0.034	0.518±0.026	0.356±0.033	0.444±0.026	0.324±0.031	0.707±0.009
	LTMSC	0.619±0.000	0.684±0.000	0.533±0.000	0.586±0.000	0.516±0.000	0.678±0.000
	t-SVD-MSC	0.798±0.000	0.811±0.000	0.729±0.000	0.757±0.000	0.788±0.000	0.728±0.000
	ULTLSE	**0.870±0.027**	**0.911±0.016**	**0.837±0.025**	**0.854±0.026**	**0.788±0.031**	**0.933±0.021**
Caltech-20	SPC_{best}	0.424±0.010	0.540±0.006	0.310±0.007	0.367±0.007	0.719±0.011	0.247±0.005
	LRR_{best}	0.522±0.033	0.545±0.008	0.380±0.037	0.440±0.037	0.739±0.019	0.314±0.035
	AMGL	0.515±0.031	0.521±0.038	0.269±0.038	0.404±0.028	0.359±0.045	0.472±0.058
	MLAN	0.530±0.007	0.473±0.002	0.202±0.007	0.376±0.007	0.281±0.003	0.571±0.018
	MCGC	0.537±0.000	0.586±0.000	0.392±0.000	0.480±0.000	0.541±0.000	0.430±0.000
	ECMSC	0.496±0.006	0.576±0.006	0.393±0.015	0.459±0.013	0.689±0.034	0.344±0.010
	CSMSC	0.533±0.037	0.605±0.009	0.421±0.003	0.480±0.040	0.770±0.020	0.349±0.037
	MSC-IAS	0.542±0.019	0.536±0.014	0.412±0.025	0.489±0.024	0.610±0.022	0.409±0.031
	MCLES	0.452±0.015	0.595±0.029	0.226±0.031	0.333±0.23	0.392±0.017	0.290±0.037
	LTMSC	0.529±0.047	0.598±0.021	0.419±0.050	0.476±0.049	0.788±0.036	0.341±0.043
	t-SVD-MSC	0.613±0.029	0.722±0.010	0.486±0.032	0.537±0.031	0.385±0.029	**0.878±0.013**
	ULTLSE	**0.670±0.022**	**0.840±0.012**	**0.636±0.018**	**0.680±0.015**	**0.910±0.010**	0.543±0.016
Caltech-101	SPC_{best}	0.484±0.019	0.723±0.032	0.319±0.014	0.340±0.025	0.597±0.018	0.235±0.020
	LRR_{best}	0.510±0.009	0.728±0.014	0.304±0.017	0.339±0.008	0.627±0.012	0.231±0.010
	AMGL	0.221±0.011	0.347±0.035	0.263±0.058	0.025±0.020	0.074±0.017	0.042±0.011
	MLAN	0.598±0.010	0.731±0.018	0.251±0.028	0.282±0.025	0.194±0.032	0.539±0.047
	MCGC	0.355±0.000	0.428±0.000	0.038±0.000	0.087±0.000	0.048±0.000	0.474±0.000
	ECMSC	0.352±0.011	0.581±0.006	0.262±0.017	0.275±0.017	0.436±0.024	0.201±0.013
	CSMSC	0.603±0.021	0.808±0.007	0.432±0.026	0.442±0.026	0.710±0.026	0.321±0.022
	MSC-IAS	0.569±0.011	0.763±0.014	0.386±0.054	0.404±0.051	0.403±0.023	0.412±0.028
	MCLES	-	-	-	-	-	-
	LTMSC	0.565±0.010	0.787±0.006	0.401±0.008	0.411±0.008	0.694±0.015	0.292±0.006
	t-SVD-MSC	0.607±0.005	0.858±0.003	0.430±0.005	0.440±0.010	**0.742±0.007**	0.323±0.009
	ULTLSE	**0.696±0.009**	**0.877±0.004**	**0.578±0.025**	**0.590±0.026**	0.549±0.018	**0.639±0.022**



(a) 3Sources (b) BBCnews (c) WikipediaArticles

(d) Scene-15 (e) ALOI (f) Caltech-20

Fig. 5: ACC of the proposed ULTLSE given various combinations of $\lambda \in \{0.01, 0.02, \ldots, 1\}$ and $\alpha \in \{0.1, 0.2, \ldots, 1\}$.

TABLE VII: Running time (seconds) on eight datasets for ten multi-view methods.

Methods	AMGL	MLAN	MCGC	ECMSC	CSMSC	MSC-IAS	MCLES	LTMSC	t-SVD-MSC	ULTLSE
3Sources	0.13	0.20	0.80	107.62	20.77	12.48	7.37	14.77	9.61	5.04
BBCnews	1.69	3.26	3.68	1231.65	181.36	36.78	4205.90	175.75	190.12	61.80
WikipediaArticles	6.99	0.99	3.01	21.41	19.29	4.60	1988.10	32.50	8.23	5.30
Scene-15	1355.86	319.04	702.51	5968.76	6423.50	344.87	-	8928.62	3085.64	1238.90
MITIndoor	865.98	746.28	2029.80	9530.68	9105.9	678.03	-	10674.58	4824.69	2725.3
ALOI	12.50	2.06	5.83	135.41	90.51	14.48	12180.63	131.56	55.87	94.46
Caltech-20	90.42	74.36	133.78	880.77	865.91	140.04	93754.21	1968.00	843.05	301.97
Caltech-101	5549.70	1254.70	7065.12	25708.86	17069.29	689.71	-	24962.91	13685.87	9983.85

TABLE V: Comparison of clustering performance (ACC/NMI) on individual view via spectral clustering.

Datasets	WikipediaArticles	Scene-15	Caltech-20
View 1	0.198/0.655	0.475/0.452	0.257/0.247
View 2	0.552/0.519	0.359/0.359	0.291/0.346
View 3	-	0.312/0.295	0.287/0.343
View 4	-	-	0.424/0.540
View 5	-	-	0.404/0.496
View 6	-	-	0.344/0.447

TABLE VI: Computational complexity of different methods.

Method	Category	Computational complexity
AMGL	garph-based	$\mathcal{O}(tcn^2)$
MLAN	garph-based	$\mathcal{O}(n^3+tcn^2)$
MCGC	garph-based	$\mathcal{O}(tmn^2)$
ECMSC	subspace-based	$\mathcal{O}((t+m)n^3)$
CSMSC	subspace-based	$\mathcal{O}(tmn^3)$
MSC-IAS	subspace-based	$\mathcal{O}(tn^2)$
MCLES	subspace-based	$\mathcal{O}(t(d^2+n^3+cn^2))$
LTMSC	tensor-based	$\mathcal{O}(tmn^3)$
t-SVD-MSC	tensor-based	$\mathcal{O}(n^3+tmn^2\log(n))$
ULTLSE	tensor-based	$\mathcal{O}(tmn^2log(n))$

consider and measure the specific contributions of different views.

3) Parameter selection and sensitivity analysis. In the model, two main parameters play important roles: λ, α. Specifically, λ is set 0.1 on the datasets 3Sources, WikipediaArticles, and ALOI, 0.01 on the datasets BBCnews, MITIndoor-67, Caltech-20, and Caltech-101, and 0.05 on the dataset Scene-15. α is set 0.8 on the datasets 3Sources and MITIndoor-67, 0.1 on the datasets WikipediaArticles and Caltech-101, 1, 0.5, 0.6, 0.4 on the datasets BBCnews, Scenes-15, ALOI, and Caltech-20, respectively. As for the parameters μ and ρ, we set their values to 10^{-2} on the datasets BBCnews, Scene-15, and MITIndoor-67. For the dataset 3Sources, Caltech-20, and Caltech101, their values are set 10^{-3}, 10^{-2}, respectively. For the dataset WikipediaArticles, their values are set 10^{-3}, 10^{-1}, respectively. For the dataset ALOI, their values are set 10^{-4}, 10^{-3}, respectively.

Furthermore, we investigate the influence of parameters λ and α on the model ULTLSE. Specifically, λ ranges in {0.01, 0.02, 0.04, 0.06, 0.1, 0.2, 0.4, 0.6, 0.8, 1} and α ranges in {0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1}. Fig. 5 shows the specific impact of different combinations of λ and α on the performance of ULTLSE on six datasets. It can be seen that the performance fluctuates relatively large for various settings of λ and α on the datasets BBCnews and Caltech-20. Especially, λ should be set relatively small on the dataset BBCnews, otherwise the performance will degrade dramatically. For the other datasets, the model is relatively insensitive to the changes of two parameters.

4) Complexity analysis and comparison. For the proposed model ULTLSE, the major computation complexity lies in the calculation of $\mathbf{E}$, $\boldsymbol{\mathcal{H}}$, and $\mathbf{F}$. Specifically, solving the subproblem of $\mathbf{E}$ needs $\mathcal{O}(mn^2)$, and updating $\mathbf{F}$ takes cn^2. As for the solution of $\boldsymbol{\mathcal{H}}$, it is the most important part and costs $\mathcal{O}(m^2n^2 + mn^2log(n))$, where the former corresponds to singular value decomposition and the latter corresponds to FFT operation and inverse FFT operation. Generally, the overall computational complexity of ULTLSE is $\mathcal{O}(t(mn^2+m^2n^2+mn^2log(n)+cn^2))$, where t represents the number of iterations and c denotes the number of clusters. However, there are situations that $n \gg m$ and $n \gg c$ in datasets. Hence, we can use $\mathcal{O}(tmn^2log(n))$ to represent the computational complexity of ULTLSE.

Table VI reveals the computational complexity of the compared multi-view algorithms. Moreover, the running time on eight datasets for ten multi-view methods is exhibited in Table VII. It is observed that graph-based approaches perform better than subspace-based and tensor-based approaches in terms of operation efficiency, which is because the self-representation learning and the tensor singular value decomposition are more time-consuming. Nevertheless, the running time of MSC-IAS is comparable to that of graph-based methods, and even shorter on the dataset Caltech-101. The main reason is that the sparse analysis technology is employed on the Laplacian matrix in the model. Additionally, it is worth noting that ULTLSE has lower computation cost compared with LTMSC and t-SVD-MSC, which benefits from the integration of low-rank tensor learning and spectral embedding.

5) Convergence analysis. When there are three or more block variables, it is unclear whether inexact ALM is convergent [62]. ULTLSE has five blocks variables and its objective function is not smooth, making it difficult to testify the convergence of ULTLSE. Nevertheless, according to the work [62], two factors are enough (but may not be necessary) for ULTLSE to be convergent: (1) Each feature matrix $\mathbf{X}^{(v)}$ is of full column rank; (2) The optimality gap produced in each iteration step is monotonically decreasing. The first factor can be satisfied by factorizing $\mathbf{Z}^{(v)}$ into $\mathbf{Q}^{(v)}\hat{\mathbf{Z}}^{(v)}$, and $\mathbf{Q}^{(v)}$ can be solved via orthogonalizing the columns of $\mathbf{X}^{(v)^T}$. As for the second factor, since the Lagrangian function Eq. (10) is convex, the monotonically decreasing factor can be realized to a certain extent according to [62]. Generally speaking, the proposed ULTLSE guarantees good

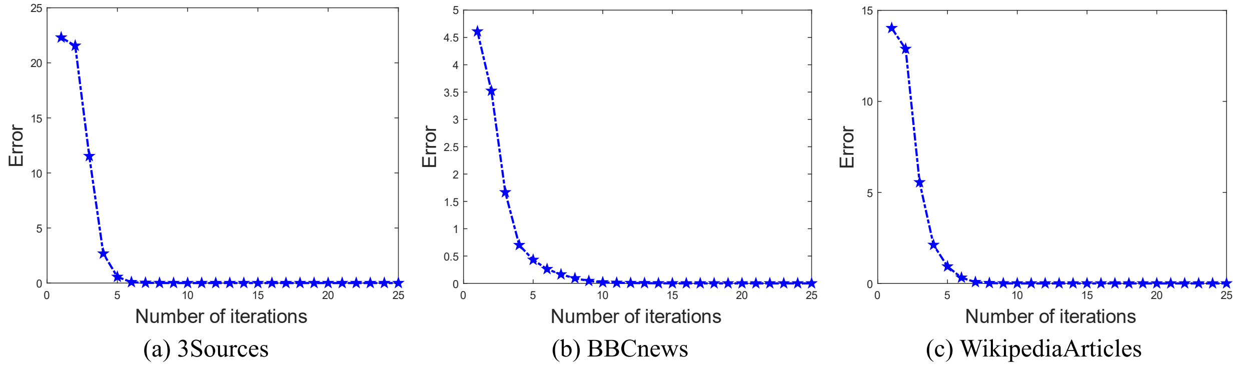


Fig. 6: Convergence results on the datasets 3Sources, BBCnews, and WikipediaArticles.

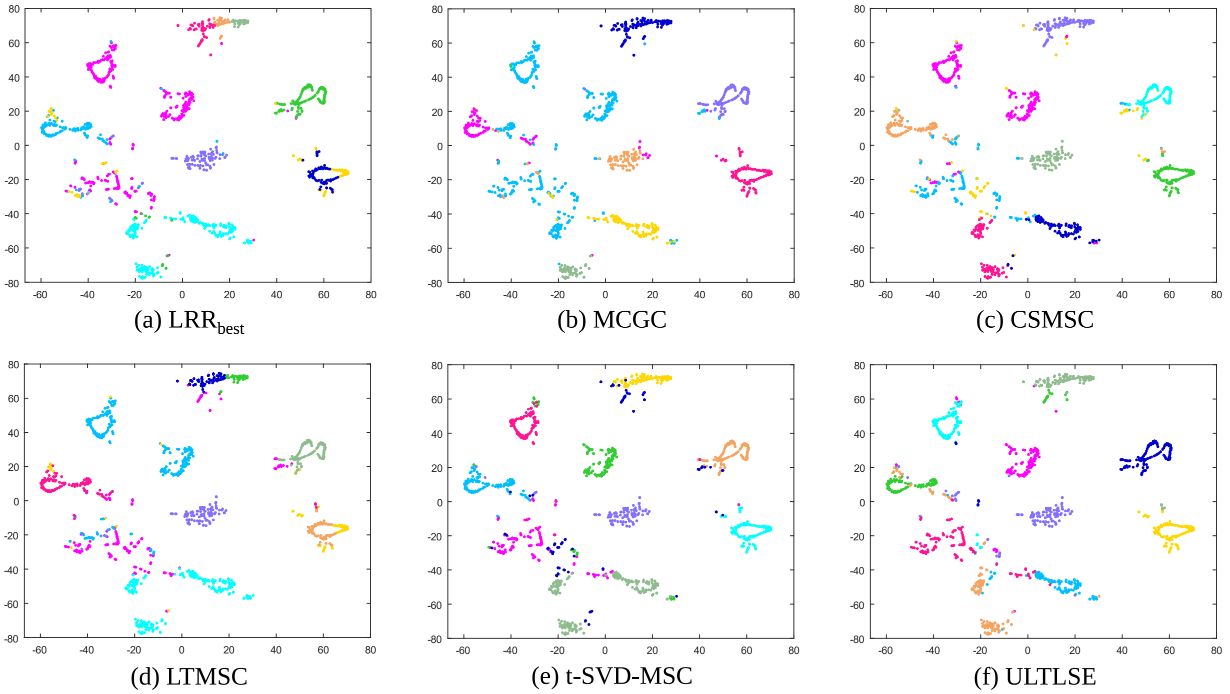


Fig. 7: The visualization of six multi-view clustering methods via t-SNE on the dataset ALOI, where different colors represent different clusters.

convergence properties. We present the empirical convergence analysis on the datasets 3Sources, BBCnews, and WikipediaArticles in Fig. 6, where the blue line represents the error of $max(\sum_{v=1}^{m}||\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}_{k+1}^{(v)} - \mathbf{E}_{k+1}^{(v)}||_\infty, \sum_{v=1}^{m}||\mathbf{Z}_{k+1}^{(v)} - \mathbf{H}_{k+1}^{(v)}||_\infty, ||\mathbf{F}_{k+1} - \mathbf{F}_k||_\infty)$. We can see the error values will converge within a certain number of iterations.

6) Visualization of clustering results. For illustrating the clustering results visually, we select the best results in each algorithm category and present them in Fig. 7, where different colors indicate different clusters. In the process, t-SNE technology is adopted to reduce the dimension of data to two. Obviously, the cluster partitions of the dataset ALOI by the algorithms LRR$_{\text{best}}$, MCGC, LTMSC are relatively messy, while CSMSC, t-SVD-MSC, and our method ULTLSE achieve the segmentation of data points well. Moreover, it can be seen that ULTLSE realizes the distinction of clusters more clearly and reasonably, which is also verified by the values of their corresponding specific evaluation metrics.

VI. Conclusion and Future Work

In this paper, we propose a novel unified framework ULTLSE for multi-view subspace clustering, which integrates optimization of low-rank tensor learning and spectral embedding. In ULTLSE, the self-representations of all views constitute the target tensor, and the t-SVD based tensor nuclear norm is adopted to recover the fundamental components. Therefore, the exploration of information complementarity and consistency is realized based on global perspective instead of pairwise matrices. While recovering a low-rank tensor space, we manage to learn the label indicator matrix by spectral embedding at the same time. The diversity of different views is also considered, an adaptive weighting coefficient is assigned to each view. Moreover, we perform extensive comparative experiments on eight real-world datasets and prove the superiority of ULTLSE. In the future, we wish to further reduce the computation complexity of the proposed ULTLSE, thereby improving the operation efficiency on large-scale datasets.

REFERENCES

[1] R. Vidal, "Subspace clustering," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 52–68, 2011.

[2] H. Kriegel, P. Kröger, and A. Zimek, "Subspace clustering," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 2, no. 4, pp. 351–364, 2012.

[3] L. Parsons, E. Haque, and H. Liu, "Subspace clustering for high dimensional data: a review," *Acm Sigkdd Explorations Newsletter*, vol. 6, no. 1, pp. 90–105, 2004.

[4] X. Shu, J. Tang, G.-J. Qi, Z. Li, Y.-G. Jiang, and S. Yan, "Image classification with tailored fine-grained dictionaries," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 2, pp. 454–467, 2018.

[5] C. Zhang, H. Fu, Q. Hu, X. Cao, Y. Xie, D. Tao, and D. Xu, "Generalized latent multi-view subspace clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 1, pp. 86–99, 2020.

[6] A. Djelouah, J. Franco, E. Boyer, F. L. Clerc, and P. Pérez, "Multi-view object segmentation in space and time," in *IEEE International Conference on Computer Vision*, pp. 2640–2647, 2013.

[7] A. Djelouah, J. Franco, E. Boyer, F. L. Clerc, and P. Pérez, "Sparse multi-view consistency for object segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1890–1903, 2015.

[8] J. Sun, J. Bi, and H. R. Kranzler, "Multi-view biclustering for genotype-phenotype association studies of complex diseases," in *Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine*, pp. 316–321, 2013.

[9] G. Chao, J. Sun, J. Lu, A. Wang, D. D. Langleben, C. R. Li, and J. Bi, "Multi-view cluster analysis with incomplete data to understand treatment effects," *Information Sciences*, vol. 494, pp. 278–293, 2019.

[10] X. Zhang, Y. Yang, T. Li, Y. Zhang, H. Wang, and H. Fujita, "CMC: A consensus multi-view clustering model for predicting alzheimer's disease progression," *Computer Methods and Programs in Biomedicine*, vol. 199, p. 105895, 2021.

[11] A. Kumar, P. Rai, and H. D. III, "Co-regularized multi-view spectral clustering," in *Advances in Neural Information Processing Systems*, pp. 1413–1421, 2011.

[12] Y. Kim, M. Amini, C. Goutte, and P. Gallinari, "Multi-view clustering of multilingual documents," in *Proceeding of the International ACM SIGIR Conference on Research and Development in Information Retrieval*.

[13] M. Chen, L. Huang, C. Wang, and D. Huang, "Multi-view clustering in latent embedding space," in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 3513–3520, 2020.

[14] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2765–2781, 2013.

[15] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 171–184, 2013.

[16] S. Luo, C. Zhang, W. Zhang, and X. Cao, "Consistent and specific multi-view subspace clustering," in *Proceedings of the International Joint Conferences on Artificial Intelligence Organization*, pp. 3730–3737, 2018.

[17] Z. Kang, W. Zhou, Z. Zhao, J. Shao, M. Han, and Z. Xu, "Large-scale multi-view subspace clustering in linear time," in *Proceedings of the International Joint Conferences on Artificial Intelligence Organization*, pp. 4412–4419, 2020.

[18] Z. Yang, Q. Xu, W. Zhang, X. Cao, and Q. Huang, "Split multiplicative multi-view subspace clustering," *IEEE Transactions on Image Processing*, vol. 28, no. 10, pp. 5147–5160, 2019.

[19] C. Zhang, H. Fu, S. Liu, G. Liu, and X. Cao, "Low-rank tensor constrained multiview subspace clustering," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1582–1590, 2015.

[20] Y. Xie, D. Tao, W. Zhang, Y. Liu, L. Zhang, and Y. Qu, "On unifying multi-view self-representations for clustering by tensor multi-rank minimization," *International Journal of Computer Vision*, vol. 126, no. 11, pp. 1157–1179, 2018.

[21] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, and S. Yan, "Tensor robust principal component analysis: Exact recovery of corrupted low-rank tensors via convex optimization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5249–5257, 2016.

[22] Q. Gao, W. Xia, Z. Wan, D. Xie, and P. Zhang, "Tensor-svd based graph learning for multi-view subspace clustering," in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 3930–3937, 2020.

[23] J. Wu, Z. Lin, and H. Zha, "Essential tensor learning for multi-view spectral clustering," *IEEE Transactions on Image Processing*, vol. 28, no. 12, pp. 5910–5922, 2019.

[24] Y. Chen, X. Xiao, and Y. Zhou, "Jointly learning kernel representation tensor and affinity matrix for multi-view clustering," *IEEE Transactions on Multimedia*, vol. 22, no. 8, pp. 1985–1997, 2020.

[25] M. E. Kilmer, K. Braman, N. Hao, and R. C. Hoover, "Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging," *SIAM Journal on Matrix Analysis and Applications*, vol. 34, no. 1, pp. 148–172, 2013.

[26] Y. Wang, L. Wu, X. Lin, and J. Gao, "Multi-view spectral clustering via structured low-rank matrix factorization," *CoRR*, vol. 29, pp. 4833–4843, 2018.

[27] S. Chang, G. Qi, C. C. Aggarwal, J. Zhou, M. Wang, and T. S. Huang, "Factorized similarity learning in networks," in *Proceedings of the IEEE International Conference on Data Mining*, pp. 60–69, 2014.

[28] J. Tang, X. Shu, G. Qi, Z. Li, M. Wang, S. Yan, and R. C. Jain, "Tri-clustered tensor completion for social-aware image tag refinement," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 8, pp. 1662–1674, 2017.

[29] J. Tang, X. Shu, Z. Li, Y. Jiang, and Q. Tian, "Social anchor-unit graph regularized tensor completion for large-scale image retagging," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 8, pp. 2027–2034, 2019.

[30] L. Wu and Y. Wang, "Robust hashing for multi-view data: Jointly learning low-rank kernelized similarity consensus and hash functions," *Image and Vision Computing*, vol. 57, pp. 58–66, 2017.

[31] L. Fu, Z. Chen, S. Huang, S. Huang, and S. Wang, "Multi-view learning via low-rank tensor optimization," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, pp. 1–6, 2021.

[32] G. Qi, W. Liu, C. C. Aggarwal, and T. S. Huang, "Joint intermodal and intramodal label transfers for extremely rare or unseen classes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 7, pp. 1360–1373, 2017.

[33] S. E. Hajjar, F. Dornaika, and F. Abdallah, "Multi-view spectral clustering via constrained nonnegative embedding," *Information Fusion*, vol. 78, pp. 209–217, 2022.

[34] J. Xie, R. B. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *Proceedings of the International Conference on Machine Learning* (M. Balcan and K. Q. Weinberger, eds.), vol. 48, pp. 478–487, 2016.

[35] Q. Wang, J. Cheng, Q. Gao, G. Zhao, and L. Jiao, "Deep multi-view subspace clustering with unified and discriminative learning," *IEEE Transactions on Multimedia*, vol. 23, pp. 3483–3493, 2021.

[36] J. Cai, S. Wang, C. Xu, and W. Guo, "Unsupervised deep clustering via contractive feature representation and focal loss," *Pattern Recognition*, vol. 123, p. 108386, 2022.

[37] M. Cheng, L. Jing, and M. K. Ng, "Tensor-based low-dimensional representation learning for multi-view clustering," *IEEE Transactions on Image Processing*, vol. 28, no. 5, pp. 2399–2414, 2019.

[38] J. Tan, Y. Shi, Z. Yang, C. Wen, and L. Lin, "Unsupervised multi-view clustering by squeezing hybrid knowledge from cross view and each view," *IEEE Transactions on Multimedia*, vol. 23, pp. 2943–2956, 2021.

[39] L. Fu, P. Lin, A. V. Vasilakos, and S. Wang, "An overview of recent multi-view clustering," *Neurocomputing*, vol. 402, pp. 148–161, 2020.

[40] Y. Chen, X. Xiao, Z. Hua, and Y. Zhou, "Adaptive transition probability matrix learning for multiview spectral clustering," *IEEE Transactions on Neural Networks and Learning Systems*, 2021. doi: 10.1109/TNNLS.2021.3059874.

[41] S. Wang and W. Guo, "Sparse multigraph embedding for multimodal feature representation," *IEEE Transactions on Multimedia*, vol. 19, no. 7, pp. 1454–1466, 2017.

[42] Y. Wang, W. Zhang, L. Wu, X. Lin, and X. Zhao, "Unsupervised metric fusion over multiview data by graph random walk-based cross-view diffusion," *IEEE Transactions on Neural Networks Learning Systems*, vol. 28, no. 1, pp. 57–70, 2017.

[43] C. Tang, X. Zheng, X. Liu, W. Zhang, J. Zhang, J. Xiong, and L. Wang, "Cross-view locality preserved diversity and consensus learning for multi-view unsupervised feature selection," *IEEE Transactions on Knowledge and Data Engineering*, 2021. doi=10.1109/TKDE.2020.3048678.

[44] C. Tang, J. Chen, X. Liu, M. Li, P. Wang, M. Wang, and P. Lu, "Consensus learning guided multi-view unsupervised feature selection," *Knowledge Based Systems*, vol. 160, pp. 49–60, 2018.

[45] F. Nie, G. Cai, and X. Li, "Multi-view clustering and semi-supervised classification with adaptive neighbours," in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 2408–2414, 2017.

[46] K. Zhan, F. Nie, J. Wang, and L. Yang, "Multiview consensus graph clustering," *IEEE Transactions on Image Processing*, vol. 28, no. 3, pp. 1261–1270, 2019.

[47] Z. Hu, F. Nie, R. Wang, and X. Li, "Multi-view spectral clustering via integrating nonnegative embedding and spectral embedding," *Information Fusion*, vol. 55, pp. 251–259, 2020.

[48] C. Tang, X. Liu, X. Zhu, E. Zhu, Z. Luo, L. Wang, and W. Gao, "CGD: multi-view clustering via cross-view graph diffusion," in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 5924–5931, 2020.

[49] Y. Wang and L. Wu, "Beyond low-rank representations: Orthogonal clustering basis reconstruction with optimized graph structure for multi-view spectral clustering," *Neural Networks*, vol. 103, pp. 1–8, 2018.

[50] Y. Chen, X. Xiao, C. Peng, G. Lu, and Y. Zhou, "Low-rank tensor graph learning for multi-view subspace clustering," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[51] X. Wang, X. Guo, Z. Lei, C. Zhang, and S. Z. Li, "Exclusivity-consistency regularized multi-view subspace clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 923–931, 2017.

[52] X. Zhang, H. Sun, Z. Liu, Z. Ren, Q. Cui, and Y. Li, "Robust low-rank kernel multi-view subspace clustering based on the schatten p-norm and correntropy," *Information Sciences*, vol. 477, pp. 430–447, 2019.

[53] C. Tang, X. Zhu, X. Liu, M. Li, P. Wang, C. Zhang, and L. Wang, "Learning a joint affinity graph for multiview subspace clustering," *IEEE Transactions on Multimedia*, vol. 21, no. 7, pp. 1724–1736, 2019.

[54] Y. Chen, S. Wang, C. Peng, Z. Hua, and Y. Zhou, "Generalized nonconvex low-rank tensor approximation for multi-view subspace clustering," *IEEE Transactions on Image Processing*, vol. 30, pp. 4022–4035, 2021.

[55] K. Chaudhuri, S. M. Kakade, K. Livescu, and K. Sridharan, "Multi-view clustering via canonical correlation analysis," in *Proceedings of the International Conference on Machine Learning*, pp. 129–136, 2009.

[56] Y. Luo, D. Tao, K. Ramamohanarao, C. Xu, and Y. Wen, "Tensor canonical correlation analysis for multi-view dimension reduction," *IEEE Transactions Knowledge Data Engineering*, vol. 27, no. 11, pp. 3111–3124, 2015.

[57] L. Houthuys, R. Langone, and J. A. K. Suykens, "Multi-view kernel spectral clustering," *Information Fusion*, vol. 44, pp. 46–56, 2018.

[58] O. Semerci, N. Hao, M. E. Kilmer, and E. L. Miller, "Tensor-based formulation and nuclear norm regularization for multienergy computed tomography," *IEEE Transactions on Image Processing*, vol. 23, no. 4, pp. 1678–1693, 2014.

[59] W. Hu, D. Tao, W. Zhang, Y. Xie, and Y. Yang, "The twist tensor nuclear norm for video completion," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 12, pp. 2961–2973, 2016.

[60] F. Nie, J. Li, and X. Li, "Parameter-free auto-weighted multiple graph learning: A framework for multiview clustering and semi-supervised classification," in *Proceedings of the International Joint Conferences on Artificial Intelligence Organization*, pp. 1881–1887, 2016.

[61] X. Wang, Z. Lei, X. Guo, C. Zhang, H. Shi, and S. Z. Li, "Multi-view subspace clustering with intactness-aware similarity," *Pattern Recognition*, vol. 88, pp. 50–63, 2019.

[62] J. Eckstein and D. P. Bertsekas, "On the douglasrachford splitting method and the proximal point algorithm for maximal monotone operators," *Mathematical Programming*, vol. 55, no. 1, pp. 293–318, 1992.

Lele Fu received his B.S. degree and M.E. degree from the College of Mathematics and Computer Science, Fuzhou University, Fuzhou, China in 2019 and 2022, respectively. He is currently pursuing the Ph.D. degree with the School of Systems Science and Engineering, Sun Yat-sen University, Guangzhou, China. His current research interests include machine learning and multi-view learning.

Zhaoliang Chen received his B.S. degree from the College of Mathematics and Computer Science, Fuzhou University, Fuzhou, China in 2019. He is currently pursuing the Ph.D. degree with the College of Computer and Data Science, Fuzhou University, Fuzhou, China. His current research interests include machine learning, deep learning, graph neural networks and recommender systems.

Yongyong Chen received the B.S. and M.S. degrees from the Shandong University of Science and Technology, Qingdao, China, in 2014 and 2017, respectively, and the Ph.D. degree from the University of Macau, Macau, in 2020. He is currently an Assistant Professor with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China. He has published more than 30 research papers in top-tier journals and conferences, including IEEE TIP, IEEE TNNLS, IEEE TMM, IEEE TCSVT, IEEE TGRS, IEEE TCI, IEEE JSTSP, Pattern Recognition and ACM MM. His research interests include image processing, data mining, and computer vision.

Shiping Wang received his Ph.D. degree from the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China in 2014. He worked as a research fellow in Nanyang Technological University from August 2015 to August 2016. He is currently a Professor with the College of Computer and Data Science, Fuzhou University, Fuzhou, China. His research interests include machine learning, computer vision and granular computing.