Unified One-step Multi-view Spectral Clustering

Chang Tang, Zhenglai Li, Jun Wang, Xinwang Liu, Wei Zhang, En Zhu

Abstract—Multi-view spectral clustering, which exploits the complementary information among graphs of diverse views to obtain superior clustering results, has attracted intensive attention recently. However, most existing multi-view spectral clustering methods obtain the clustering partitions in a two-step scheme, i.e., spectral embedding and subsequent k-means. This two-step scheme inevitably seeks sub-optimal clustering results due to the information loss during the two-steps processes. Besides, existing multi-view spectral clustering methods do not jointly utilize the information of graphs and embedding matrices, which also degrades final clustering results. To solve these issues, we propose a unified one-step multi-view spectral clustering method, which integrates the spectral embedding and k-means into a unified framework to obtain discrete clustering labels with a one-step strategy. Under the observation that the inner product of the embedding matrix is a low-rank approximation of the graph, we combine graphs and embedding matrices of different views to obtain a unified graph. Then, we directly capture the discrete clustering indicator matrix from the unified graph. Furthermore, we design an effective optimization algorithm to solve the resultant problem. Finally, a set of experiments on various datasets are conducted to verify the effectiveness of the proposed method. The demo code of this work is publicly available at https://github.com/guanyuezhen/UOMvSC.

Index Terms—Multi-view learning, clustering, spectral embedding, multi-view spectral clustering.

1 Introduction

Clustering algorithms aim to partition data into their groups in an unsupervised manner. As a useful tool to analyze data structure, clustering algorithms have been widely studied and employed in various fields, such as documents clustering [1], image retrieval [2], community detection in social networks [3], [4], and single-cell RNA-seq data clustering [5]. With the development of the information technique, data can be easily obtained from various sources or different feature descriptors [6], [7], [8], [9]. For example, image and language are jointly explored in vision and language understanding tasks [10]. The SIFT [11], HOG [12], LBP [13] and GIST [14] feature descriptors are widely used in image/video processing [15], [16]. Compared with the single view feature, the multi-view one usually gives more comprehensive information for exploring the underlying structure of data, leading to intensive studies of multi-view clustering.

Based on the way of generating partition from multi-view information, existing multi-view clustering methods can be generally divided into three categories, including matrix factorization based ones [17], [18], [19], [20], graph-based ones [21], [22], [23], [24], [25], [26], [27], [28], and deep learning-based ones [29], [30], [31]. The matrix factorization-based methods factorize data into a consensus representation to capture the complementary information among multiple views, then, obtain the clustering results upon the consensus representation. The deep learning-based methods utilize powerful presentation capabilities of the deep neural network to seek the clustering results from multi-view data. The graph-based methods also known as multi-view spectral clustering algorithms learn the data similarities from diverse views and capture the clustering results based on spectral clustering [32].

Spectral clustering, owning to its well-defined mathematical formulation and the ability to cluster data of arbitrary shape, has been intensively studied recently. The multi-view spectral clustering methods usually contain two main stages: 1) constructing similarity graphs from multi-view data; 2) conducting the spectral clustering on the similarity graphs to obtain the final partition results. The purpose of the first stage is to explore the neighborhood correlation between data samples. Due to diverse sources of data acquisition, the features of multi-view data are of redundancy, correlation, and diversity [33]. Thus, how to effectively explore the information of multi-view data to construct the similarity graph and boost the clustering performance becomes a critical problem in multi-view spectral clustering tasks. The second stage mainly focuses on how to obtain more accurate discrete clustering results from the pre-computed similarity graphs. For example, Abhishek et al. [34] employed a co-regularizing strategy to capture the consensus underlying clustering structure of multiple graphs under a spectral clustering framework.

Although existing multi-view clustering methods can achieve promising performance, we observe that they still suffer from two drawbacks. First, most of them employ

- *C. Tang, Z. Li and J. Wang are with the school of computer, China University of Geosciences, Wuhan 430074, China. C. Tang is also with State Key Lab. for Novel Software Technology, Nanjing University, Nanjing 210023, China. E-mail: {tangchang, yuezhenguan, wang_jun}@cug.edu.cn.*
- *X. Liu and En Zhu are with the school of computer, National University of Defense Technology, Changsha 410073, China. E-mail: {xinwangliu, enzhu}@nudt.edu.cn.*
- *W. Zhang is with Shandong Provincial Key Laboratory of Computer Networks, Shandong Computer Science Center (National Supercomputing Center in Jinan), Qilu University of Technology (Shandong Academy of Sciences), Jinan 250000, China. E-mail: wzhang@qlu.edu.cn.*

Manuscript received April 19, 2005; revised August 26, 2015. This work was supported in part by the National Science Foundation of China under Grant 62076228, and in part by Natural Natural Science Foundation of Shandong Province under grant ZR2021LZH001, and in part by Opening Fund of State Key Lab. for Novel Software Technology, Nanjing University under Grant KFKT2021B24. (Corresponding authors: Xinwang Liu and Wei Zhang). Chang Tang and Zhenglai Li contribute equally as first author to this work.

a two-step approach to obtain the discrete clustering results from the pre-computed multi-view similarity graphs, including, spectral embedding and subsequent k-means. Such a two-step pipeline leads to sub-optimal result quality due to the information loss during the two-step procedure. Second, they do not jointly utilize the information of graphs and embedding matrices to obtain the clustering results. Considering the inner product of an embedding matrix is a low-rank approximation of similarity graph (discussed in Section 4.1), it contains partial clustering structure, but less noise information of similarity graph. Thus, a combination of the inner product of the embedding matrix and similarity graph can provide more accurate clustering results.

To solve the above issues, we propose UOMvSC (short for Unified One-step Multi-view Spectral Clustering), a novel solution that integrates multiple spectral embedding matrices and graphs into a unified graph from which the clustering results are obtained in a one-step manner. Specifically, UOMvSC linearly combines multi-view embedding matrices and graphs to formulate a unified graph, in which the clustering structure is enhanced and the noise information is suppressed. Then, we directly obtain the clustering results from the unified graph.

The contributions are summarized as follows,

- We propose UOMvSC to incorporate the spectral clustering and k-means into one framework to obtain a unified graph, in which the information of view-specific graphs and spectral embedding matrices are jointly exploited. As a result, the block-diagonal structure of the similarity graph is enhanced, while the inaccurate similarities are suppressed.
- We directly obtain the discrete clustering indicators from the unified graph without other post-processing. The one-step scheme can reduce information loss and obtain better clustering results.
- To solve the resultant problem, we partially relax the clustering indicators into continuous values to formulate an equivalent optimization problem and design an effective optimization algorithm to solve the resultant model. Extensive experiments on various benchmark datasets are conducted to demonstrate the efficacy of the proposed method.

The rest of this paper is organized as follows. Section 2 gives a brief review of the most related work. In section 3, we present the preliminaries, including the formulations of spectral clustering and k-means. The details of the proposed method and designed optimization algorithm are introduced in Section 4. Section 5 provides a series of experimental results and discussions. In Section 6, we provide a conclusion of this paper.

2 Related Work

In this section, we give a brief review of existing multi-view clustering methods which can be generally grouped into three categories, i.e., matrix factorization based ones [17], [19], [20], graph-based ones [21], [22], [23], [24], [26], and deep learning-based ones [29], [30], [31].

The matrix factorization-based methods factorize data into a consensus representation to capture the complementary information among multiple views, then, obtain the clustering results upon the consensus representation. Liu et al. [35] proposed a matrix factorization-based multi-view clustering method by seeking a factorization that provides compatible solutions across diverse views. Based on deep matrix factorization [36], Zhao et al. [20] progressively factorized the multi-view data into a consensus latent space to capture the complementary information among multiple views. Meanwhile, a Laplacian regularization is introduced to preserve the locality of data in the latent space. Considering the data distribution in the learned subspace, Yang et al. [19] proposed a tri-factorization-based non-negative matrix factorization model to learn decompositions in a uniform distribution to boost the separability of the learned representation.

The deep learning-based methods utilize the powerful presentation capabilities of the deep neural network to seek the clustering results from multi-view data. In [30], a consistent generative adversarial network is designed to learn a common representation from incomplete multi-view for clustering. In [31], multiple deep embedding features are learned to comprehensively depict each image. Then, a joint learning framework is proposed to simultaneously conduct feature embedding, multi-view information fusing, and data clustering. In [29], a deep multi-view clustering method with a collaborative training scheme is proposed to capture the consensus-complementary information of different views and collaboratively learn the feature presentation and cluster assignments.

The graph-based methods learn the data similarities from diverse views and capture the clustering results based on spectral clustering. The clustering performance of graph-based methods largely depends on the quality of learned similarity graphs and the effectiveness of approaches to obtain the clustering results. The self-representation and adaptive neighbor graph learning [37] are two useful approaches to construct the similarity of data samples. Base on them, various methods are proposed to learn superior graphs from multi-view data for clustering. In [38], a subspace segmentation-based method is proposed to learn a joint affinity graph from multiple views for clustering with a rank constraint and a diversity regularization. Zhang et al. [21] jointly learned the latent representation and subspace representations, and further combined the latent representation learning process with a neural network to improve the generalization. Another kind of method mainly focuses on studying how to learn superior clustering results from some pre-computed similarity graphs [39], [40].

Existing multi-view clustering methods still suffer from two drawbacks. First, employing a two-step approach to obtain the discrete clustering results from the pre-computed multi-view similarity graphs would obtain sub-optimal clustering results. Second, the information of graphs and embedding matrices are not jointly exploited in existing multi-view clustering methods. To reduce information loss, we propose a novel multi-view spectral clustering method.

3 Preliminaries

3.1 Spectral Clustering

Given a single-view dataset $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_n]^\top \in \mathrm{R}^{n \times d}$ with n and d denoting the number of samples and features,

we can construct a weighted undirected graph $\mathcal{G}(\mathbf{X}, \mathbf{S})$ to depict the correlation of samples. The goal of spectral clustering is to cut the graph into k unconnected sub-graphs $C_1, ..., C_k$. There are two widely used graph cut methods, including, Ratio-cut and Normalized-cut. They can be formulated as follows,

$$Rcut(C1, ..., C_k) = \sum_{l=1}^{k} \frac{cut(C_l, \bar{C}_l)}{|C_l|} \tag{1}$$

$$Ncut(C1, ..., C_k) = \sum_{l=1}^{k} \frac{cut(C_l, \bar{C}_l)}{vol(C_l)} \tag{2}$$

where $\bar{C}_l$ denotes the complement part of C_l, $cut(C_l, \bar{C}_l)$ cuts the connection between C_l and $\bar{C}_l$, $vol(C_l) = \sum_{i \in C_l} d_{ii}$, d_{ii} is the degree of sample $\mathbf{x}_i$ computed as $d_{ii} = \sum_{j=1}^{n} s_{ij}$.

By transforming $\sum_{l=1}^{k} \frac{cut(C_l, \bar{C}_l)}{|C_l|}$ into $\sum_{l=1}^{k} \frac{\mathbf{y}_l^\top \mathbf{L} \mathbf{y}_l}{\mathbf{y}_l^\top \mathbf{y}_l}$, Eq. (1) can be rewritten as the following trace minimizing formulation,

$$\begin{aligned} &\min_{\mathbf{Y}} \mathrm{Tr}((\mathbf{Y}^\top \mathbf{Y})^{-\frac{1}{2}} \mathbf{Y}^\top \mathbf{L} \mathbf{Y} (\mathbf{Y}^\top \mathbf{Y})^{-\frac{1}{2}}) \\ &s.t.\ \mathbf{Y} \in \{0,1\}^{n \times k}, \mathbf{y}_i \mathbf{1} = 1. \end{aligned} \tag{3}$$

where $\mathbf{Y}$ is a discrete clustering indicator matrix, $\mathbf{L} = \mathbf{D} - \mathbf{S}$ is the Laplacian matrix of graph $\mathbf{S}$, $\mathbf{D}$ represents a degree matrix.

Similar to the Ratio-cut, the Normalized-cut in Eq. (2) can be rewritten as,

$$\begin{aligned} &\min_{\mathbf{Y}} \mathrm{Tr}(\mathbf{D}^{\frac{1}{2}} (\mathbf{Y}^\top \mathbf{D} \mathbf{Y})^{-\frac{1}{2}} \mathbf{Y}^\top \mathbf{D}^{-\frac{1}{2}} \mathbf{L} \mathbf{D}^{-\frac{1}{2}} \mathbf{Y} (\mathbf{Y}^\top \mathbf{D} \mathbf{Y})^{-\frac{1}{2}} \mathbf{D}^{\frac{1}{2}}) \\ &s.t.\ \mathbf{Y} \in \{0,1\}^{n \times k}, \mathbf{y}_i \mathbf{1} = 1. \end{aligned} \tag{4}$$

The problem in Eq. (3) and Eq. (4) are hard to be optimized, since $\mathbf{Y}$ is discrete. A widely used manner is to relax the discrete $\mathbf{Y}$ into continuous values. In such a scheme, the problems in Eq. (3) and Eq. (4) can both reformulated into following optimization form,

$$\min_{\mathbf{H}} \mathrm{Tr}(\mathbf{H}^\top \bar{\mathbf{L}} \mathbf{H}) \quad s.t.\ \mathbf{H}^\top \mathbf{H} = \mathbf{I}_k \tag{5}$$

where

$$\bar{\mathbf{L}} = \begin{cases} \mathbf{D} - \mathbf{S}, & Ratio - cut \\ \mathbf{D}^{-\frac{1}{2}} (\mathbf{D} - \mathbf{S}) \mathbf{D}^{-\frac{1}{2}}, & Normalized - cut \end{cases} \tag{6}$$

The optimal solution of $\mathbf{H}$ can be obtained by taking the eigenvectors corresponding to the smallest k eigenvalues of $\bar{\mathbf{L}}$. And the final discrete clustering results can be captured by conducting spectral rotation or k-means on the spectral embedding matrix $\mathbf{H}$.

3.2 k-means

k-means clustering is a widely used method to obtain the discrete clustering indicator matrix $\mathbf{Y}$ from the spectral embedding matrix $\mathbf{H}$. And the process can be formulated as,

$$\begin{aligned} &\min_{\mathbf{Y}} \sum_{i=1}^{n} \sum_{c=1}^{k} y_{ic} \|\mathbf{h}_i - \mu_c\|_2^2 \\ &s.t.\ \mathbf{Y} \in \{0,1\}^{n \times k}, \mathbf{y}_i \mathbf{1} = 1. \end{aligned} \tag{7}$$

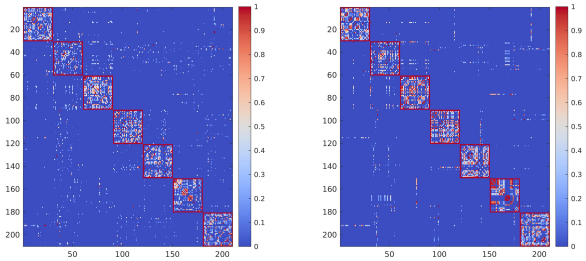
where $\mathbf{h}_i$ and μ_c are the i-th sample and c-th cluster centroid of the spectral embedding matrix, respectively. Eq. (7) can be rewritten into matrix factorization version as [41],

$$\begin{aligned} &\max_{\mathbf{Y}} \mathbf{Tr}((\mathbf{Y}^\top \mathbf{Y})^{-\frac{1}{2}} \mathbf{Y}^\top \mathbf{H} \mathbf{H}^\top \mathbf{Y} (\mathbf{Y}^\top \mathbf{Y})^{-\frac{1}{2}}) \\ &s.t.\ \mathbf{Y} \in \{0,1\}^{n \times k}, \mathbf{y}_i \mathbf{1} = 1. \end{aligned} \tag{8}$$

4 The Proposed Method

In this section, we firstly integrate the spectral clustering and k-means into a unified framework, in which the information of similarity graph and low dimensional spectral embedding matrix can be jointly explored. Then, we extend the single-view clustering framework into a multi-view version. Finally, we design an effective iteration algorithm for optimizing the resultant problem.

4.1 A Unified Learning Framework



(a) $\mathbf{A}_x$ (b) $\mathbf{A}_h$

Fig. 1. The similarity matrices constructed on MSRV1 dataset. For better visualization, the similarities are scaled into $[0, 1]$.

Given a symmetric and doubly stochastic graph $\mathbf{A}_x$ constructed from data $\mathbf{X}$ with a degree matrix $\mathbf{D}_x = \mathbf{I}$, the spectral clustering in Eq. (5) can be further rewritten as,

$$\min_{\mathbf{H}} \|\mathbf{A}_x - \mathbf{H}\mathbf{H}^\top\|_F^2 \quad s.t.\ \mathbf{H}^\top \mathbf{H} = \mathbf{I}_k. \tag{9}$$

As observed in Eq. (9), the inner product of a spectral embedding matrix $\mathbf{H}$ is a low-rank approximation of the similarity graph $\mathbf{A}_x$. Thus, the inner product of spectral embedding matrix can preserve partial clustering structure and filter some noise information of the similarity graph. And a combination of similarity graph and corresponding embedding matrix enables to provide more accurate clustering results.

By replacing the similarity matrix $\mathbf{S}$ with the symmetric and doubly stochastic graph $\mathbf{A}_x$, Eq. (3) and Eq. (4) can be both rewritten as,

$$\begin{aligned} &\max_{\mathbf{Y}} \mathbf{Tr}((\mathbf{Y}^\top \mathbf{Y})^{-\frac{1}{2}} \mathbf{Y}^\top \mathbf{A}_x \mathbf{Y} (\mathbf{Y}^\top \mathbf{Y})^{-\frac{1}{2}}) \\ &s.t.\ \mathbf{Y} \in \{0,1\}^{n \times k}, \mathbf{y}_i \mathbf{1} = 1. \end{aligned} \tag{10}$$

Since Eq. (10) and Eq. (8) have similar formulations, we can easily integrate them into a unified framework as,

$$\begin{aligned} &\max_{\mathbf{Y}} \mathbf{Tr}((\mathbf{Y}^\top \mathbf{Y})^{-\frac{1}{2}} \mathbf{Y}^\top \mathbf{A}^* \mathbf{Y} (\mathbf{Y}^\top \mathbf{Y})^{-\frac{1}{2}}) \\ &s.t.\ \mathbf{Y} \in \{0,1\}^{n \times k}, \mathbf{y}_i \mathbf{1} = 1, \\ &\quad \mathbf{A}^* = (1 - \lambda)\mathbf{A}_x + \lambda \mathbf{H}\mathbf{H}^\top. \end{aligned} \tag{11}$$

where $\mathbf{A}^*$ is a unified graph, λ is a combination coefficient satisfing $0 \leq \lambda \leq 1$. In Eq. (11), $\mathbf{H}\mathbf{H}^\top$ can be treated as a dense connected graph. Here, we adopt the adaptive neighbor graph learning approach to further filter some inaccurate sample correction and construct a similarity graph $\mathbf{A}_h$. Afterwards, Eq. (11) can be rewritten as,

$$\begin{aligned} &\max_{\mathbf{Y}} \mathbf{Tr}((\mathbf{Y}^\top\mathbf{Y})^{-\frac{1}{2}}\mathbf{Y}^\top\mathbf{A}^*\mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-\frac{1}{2}}) \\ &s.t.\ \mathbf{Y} \in \{0,1\}^{n\times k}, \mathbf{y}_i\mathbf{1} = 1, \\ &\quad \mathbf{A}^* = (1-\lambda)\mathbf{A}_x + \lambda\mathbf{A}_h. \end{aligned} \tag{12}$$

As seen in Figure 1, $\mathbf{A}_x$ is constructed from original features. Thus, $\mathbf{A}_x$ preserves the underlying clustering structure as well as mixed noise information. The spectral embedding $\mathbf{H}$ preserves the top k principal components of $\mathbf{A}_x$. As a result, the noises and some information on the clustering structure can be somewhat filtered out. In this paper, we integrate $\mathbf{A}_x$ and $\mathbf{A}_h$ into a unified similarity matrix to enhance the block-diagonal structure and suppress inaccurate similarities to obtain better clustering results.

Eq. 12 is hard to solve due to discrete constraint on $\mathbf{Y}$. Here, we partially relax $\mathbf{Y}$ into continuous values, i.e., $\mathbf{H}^{*\top}\mathbf{H}^* = \mathbf{I}_k$ and $\mathbf{H}^* \approx \mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-\frac{1}{2}}$. And Eq. (12) is transferred as,

$$\begin{aligned} &\max_{\mathbf{Y},\mathbf{H}^*} \mathbf{Tr}(\mathbf{H}^{*\top}\mathbf{A}^*\mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-\frac{1}{2}}) \\ &s.t.\ \mathbf{H}^{*\top}\mathbf{H}^* = \mathbf{I}_k, \mathbf{Y} \in \{0,1\}^{n\times k}, \mathbf{y}_i\mathbf{1} = 1, \\ &\quad \mathbf{A}^* = (1-\lambda)\mathbf{A}_x + \lambda\mathbf{A}_h, \end{aligned} \tag{13}$$

In such a formulation, the problem can be easier solved by an iteration algorithm. In each iteration, $\mathbf{Y}$ can be updated by the approach in [42], while $\mathbf{H}^*$ can be obtained by performing Singular Value Decomposition on $\mathbf{A}^*\mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-\frac{1}{2}}$. Please refer Section 4.3 for detailed solutions. We further verify that solving Eq. (12) is equal to solving Eq. (13) as follows,

Theorem 1. *maximizing Eq.* (12) *is conceptually equivalent to maximize* $\mathrm{Tr}(\mathbf{H}^{*\top}\mathbf{A}^*\mathbf{F})$ *with constraints* $\mathbf{H}^{*\top}\mathbf{H}^* = \mathbf{I}_k$ *and* $\mathbf{F} = \mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-\frac{1}{2}}$.

Proof. By taking the Singular Value Decomposition of $\mathbf{A}^*$, i,e., $\mathbf{A}^* = \mathbf{U}_a\mathbf{\Sigma}_a\mathbf{U}_a^\top$, we can see that,

$$\begin{aligned} &\mathrm{Tr}(\mathbf{H}^{*\top}\mathbf{A}^*\mathbf{F}) \\ &= \mathrm{Tr}(\mathbf{H}^{*\top}\mathbf{U}_a\mathbf{\Sigma}_a\mathbf{U}_a^\top\mathbf{F}) \\ &= \mathrm{Tr}(\mathbf{H}^{*\top}\mathbf{U}_a\mathbf{\Sigma}_a^{0.5}\mathbf{\Sigma}_a^{0.5}\mathbf{U}_a^\top\mathbf{F}) \\ &\leq \frac{1}{2}(\|\mathbf{H}^{*\top}\mathbf{U}_a\mathbf{\Sigma}_a^{0.5}\|_F^2 + \|\mathbf{F}^\top\mathbf{U}_a\mathbf{\Sigma}_a^{0.5}\|_F^2) \\ &= \frac{1}{2}(\mathrm{Tr}(\mathbf{H}^{*\top}\mathbf{A}^*\mathbf{H}^*) + \mathrm{Tr}(\mathbf{F}^\top\mathbf{A}^*\mathbf{F})) \\ &\leq \frac{1}{2}(\sum_{i=1}^{k}\sigma_i + \mathrm{Tr}(\mathbf{F}^\top\mathbf{A}^*\mathbf{F})) \end{aligned} \tag{14}$$

where $\sigma_i, i = 1, ..., k$ are the top k largest singular values of matrix $\mathbf{A}^*$. The equivalence of the inequation holds when $\mathbf{H}^* = \mathbf{F}$. As can be seen from Eq. (14), $\mathrm{Tr}(\mathbf{H}^{*\top}\mathbf{A}^*\mathbf{F})$ is a lower-bound of $\frac{1}{2}(\sum_{i=1}^{k}\sigma_i + \mathrm{Tr}(\mathbf{F}^\top\mathbf{A}^*\mathbf{F}))$. Thus, we can utilize $\mathrm{Tr}(\mathbf{H}^{*\top}\mathbf{A}^*\mathbf{F}) - 0.5\sum_{i=1}^{k}\sigma_i$ to approximate $\mathrm{Tr}(\mathbf{F}^\top\mathbf{A}^*\mathbf{F})$. Since $\sum_{i=1}^{k}\sigma_i$ can be treated as a constant, maximizing Eq.(12) is conceptually equivalent to maximize $\mathrm{Tr}(\mathbf{H}^{*\top}\mathbf{A}^*\mathbf{F})$ with constraints $\mathbf{H}^{*\top}\mathbf{H}^* = \mathbf{I}_k$ and $\mathbf{F} = \mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-\frac{1}{2}}$. □

As observed in Eq. (13), the unified graph $\mathbf{A}^* = (1-\lambda)\mathbf{A}_x + \lambda\mathbf{A}_h$ well encodes the information of similarity graph and low dimensional spectral embedding matrix.

4.2 The Proposed Multi-view Spectral Clustering

The samples are commonly described by multiple feature descriptors or captured from diverse sources. Considering this case, we extend above single-view framework into a multi-view version. Let $\{\mathbf{A}_x^{(v)}\}_{v=1}^V$ be the view-specific symmetric and doubly stochastic graphs, and $\{\mathbf{H}^{(v)}\}_{v=1}^V$ be their corresponding spectral embedding matrices. The similarity matrices $\{\mathbf{A}_h^{(v)}\}_{v=1}^V$ can be also obtained from $\{\mathbf{H}^{(v)}\}_{v=1}^V$. Then, we integrate these graphs into a unified graph which can be formulated as,

$$\begin{aligned} \mathbf{A}^* =& (1-\lambda)\sum_{v=1}^{V}\alpha_v\mathbf{A}_x^{(v)} + \lambda\sum_{v=1}^{V}\beta_v\mathbf{A}_h^{(v)} \\ &s.t. \sum_{v=1}^{V}\alpha_v^2 = 1, \alpha_v \geq 0, \sum_{v=1}^{V}\beta_v^2 = 1, \beta_v \geq 0. \end{aligned} \tag{15}$$

where α_v and β_v are the weights of v-th view graph and spectral embedding matrix, respectively. Considering that $\mathbf{A}_x^v$ and $\mathbf{A}_h^v$ depict different aspects of the sample correlations of v-th view. We assign $\mathbf{A}_x^v$ and $\mathbf{A}_h^v$ with different view weights, i.e., α_v, β_v, to fully exploit the complementary information among diverse views.

Based the unified similarity graph, we have,

$$\begin{aligned} &\max_{\mathbf{H}^*,\mathbf{Y},\alpha,\beta} \mathbf{Tr}(\mathbf{H}^{*\top}\mathbf{A}^*\mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-\frac{1}{2}}) \\ &\quad s.t. \sum_{v=1}^{V}\alpha_v^2 = 1, \alpha_v \geq 0, \sum_{v=1}^{V}\beta_v^2 = 1, \beta_v \geq 0, \\ &\quad \mathbf{Y} \in \{0,1\}^{n\times k}, \mathbf{y}_i\mathbf{1} = 1, \mathbf{H}^{*\top}\mathbf{H}^* = \mathbf{I}_k, \\ &\quad \mathbf{A}^* = (1-\lambda)\sum_{v=1}^{V}\alpha_v\mathbf{A}_x^{(v)} + \lambda\sum_{v=1}^{V}\beta_v\mathbf{A}_h^{(v)}. \end{aligned} \tag{16}$$

Eq. (16) integrates the spectral clustering and k-means into a unified framework, in which the information of the spectral embedding matrices and similarity graphs are jointly explored to generate the clustering results. Besides, The clustering indicator matrix can be directly obtained from the unfied graph without other post-processing.

4.3 Optimization

In this part, we design an effective iteration algorithm for optimizing the resultant problem in Eq. (16). By employing an alternative optimization strategy, Eq. (16) can be divided into four sub-problems including, $\mathbf{Y}$ sub-problem, $\mathbf{H}^*$ sub-problem, $\boldsymbol{\alpha}$ sub-problem and $\boldsymbol{\beta}$ sub-problem.

Update $\mathbf{Y}$: when variables $\mathbf{H}^*, \boldsymbol{\alpha}, \boldsymbol{\beta}$ are fixed, the problem in Eq. (16) can be rewritten as,

$$\begin{aligned} &\max_{\mathbf{Y}} \mathbf{Tr}(\mathbf{H}^{*\top}\mathbf{A}^*\mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-\frac{1}{2}}) \\ &s.t.\ \mathbf{Y} \in \{0,1\}^{n\times k}, \mathbf{y}_i\mathbf{1} = 1. \end{aligned} \tag{17}$$

Eq. (17) is equal to the following optimization problem,

$$\begin{aligned} &\max_{\mathbf{Y}} \mathbf{Tr}((\mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-\frac{1}{2}})^\top(\mathbf{A}^{*\top}\mathbf{H}^*)) \\ &s.t.\ \mathbf{Y}\in\{0,1\}^{n\times k}, \mathbf{y}_i\mathbf{1}=1. \end{aligned} \tag{18}$$

Taking $\mathbf{G}=\mathbf{A}^{*\top}\mathbf{H}^*$ to Eq. (18), the formulation of Eq. (18) can be transformed as,

$$\begin{aligned} &\max_{\mathbf{Y}} \mathbf{Tr}((\mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-\frac{1}{2}})^\top\mathbf{G}) \\ &=\sum_{c=1}^{k}\frac{\sum_{i=1}^{n} y_{ic}g_{ic}}{\sqrt{\mathbf{y}_c^\top\mathbf{y}_c}} \\ &s.t.\ \mathbf{Y}\in\{0,1\}^{n\times k}, \mathbf{y}_i\mathbf{1}=1. \end{aligned} \tag{19}$$

As given in Eq. (19), $\sqrt{\mathbf{y}_c^\top\mathbf{y}_c}$ involves all rows of $\mathbf{Y}$. Thus, we tend to sequentially solve $\mathbf{Y}$ row by row. Assume we have captured the optimal solution of Eq. (19) as $\bar{\mathbf{Y}}$, we only need to explore the increment of objective function values in Eq. (19), when y_{ic} changes from 0 to 1. The increment can be formulated as follows,

$$\delta_{ic}=\frac{\sum_{j=1}^{n} y_{jc}g_{jc}+g_{ic}(1-y_{ic})}{\sqrt{\mathbf{y}_c^\top\mathbf{y}_c+(1-y_{ic})}}-\frac{\sum_{j=1}^{n} y_{jc}g_{jc}-g_{ic}y_{ic}}{\sqrt{\mathbf{y}_c^\top\mathbf{y}_c-y_{ic}}} \tag{20}$$

Afterwards, i-th row of $\mathbf{Y}$ can be obtained as,

$$y_{ic}=<arg\max_{c\in[1,k]}\ \delta_{ic}> \tag{21}$$

where $<.>$ denotes 1 if the argument is true or 0 otherwise.

Update $\mathbf{H}^*$: when variables $\mathbf{Y},\boldsymbol{\alpha},\boldsymbol{\beta}$ are fixed, the problem in Eq. (16) can be rewritten as,

$$\begin{aligned} &\max_{\mathbf{H}^*} \mathbf{Tr}(\mathbf{H}^{*\top}\mathbf{A}^*\mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-\frac{1}{2}}) \\ &s.t.\ \mathbf{H}^{*\top}\mathbf{H}^*=\mathbf{I}_k. \end{aligned} \tag{22}$$

Taking $\mathbf{M}=\mathbf{A}^*\mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-\frac{1}{2}}$ to Eq. (22), the formulation of Eq. (22) can be rewritten as,

$$\max_{\mathbf{H}^*} \mathbf{Tr}(\mathbf{H}^{*\top}\mathbf{M})\quad s.t.\ \mathbf{H}^{*\top}\mathbf{H}^*=\mathbf{I}_k. \tag{23}$$

According to the Theorem in [43], the optimal solution of $\mathbf{H}^*$ in Eq. (23) can be obtained as,

$$\mathbf{H}^*=\mathbf{U}_m\mathbf{V}_m^\top \tag{24}$$

where $\mathbf{U}_m$ and $\mathbf{V}_m$ are the right and left parts of the Singular Value Decomposition of matrix $\mathbf{M}$, respectively.

Update $\boldsymbol{\alpha}$: when variables $\mathbf{Y},\mathbf{H}^*,\boldsymbol{\beta}$ in Eq. (16) are fixed, we have,

$$\max_{\alpha}\ \sum_{v=1}^{V}\alpha_v\eta_v\ \ s.t.\sum_{v=1}^{V}\alpha_v^2=1,\alpha_v\geq 0. \tag{25}$$

where $\eta_v=\mathrm{Tr}(\mathbf{H}^{*\top}\mathbf{A}_x^{(v)}\mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-\frac{1}{2}})$, the closed-form solution of α_v can be formulated as,

$$\alpha_v=\frac{\eta_v}{\sqrt{\sum_{v=1}^{V}\eta_v^2}} \tag{26}$$

Update $\boldsymbol{\beta}$: when variables $\mathbf{Y},\mathbf{H}^*,\boldsymbol{\alpha}$ in Eq. (16) are fixed, we have,

$$\max_{\beta}\ \sum_{v=1}^{V}\beta_v\gamma_v\ \ s.t.\sum_{v=1}^{V}\beta_v^2=1,\beta_v\geq 0. \tag{27}$$

where $\gamma_v=\mathrm{Tr}(\mathbf{H}^{*\top}\mathbf{A}_h^{(v)}\mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-\frac{1}{2}})$, the closed-form solution of β_v can be similarly updated as,

$$\beta_v=\frac{\gamma_v}{\sqrt{\sum_{v=1}^{V}\gamma_v^2}} \tag{28}$$

We summarized the whole procedure for solving the problem in Eq. (16) in Algorithm 1. In each iteration of Algorithm 1, we only need to update $\mathbf{Y}$ once via Eq. (21) rather than multiple times with another iteration process as [42].

4.4 Algorithm Analysis

4.4.1 Graph Construction

We adopt a adaptive graph learning method to compute the similarity matrices. Given multi-view data $\{\mathbf{X}^{(v)}\}_{v=1}^{V}$, $\mathbf{X}^{(v)}\in\mathrm{R}^{d_v\times n}$, where d_v, n, and V denote the numbers of v-th view features, samples and total views, respectively. The adaptive graph learning method can be formulated as,

$$\begin{aligned} &\min_{\mathbf{A}_x^{(v)}}\sum_{i=1}^{n}\sum_{j=1}^{n}\|\mathbf{x}_i^{(v)}-\mathbf{x}_j^{(v)}\|_2^2A_{xij}^{(v)}+\gamma A_{xij}^{(v)2} \\ &s.t.\ 0\leq A_{xij}^{(v)}\leq 1,\sum_{i=1}^{n}A_{xij}^{(v)}=1 \end{aligned} \tag{29}$$

where $\mathbf{x}_i^{(v)}$ and $\mathbf{x}_j^{(v)}$ are the i-th and j-th samples of v-th view, $A_{xij}^{(v)}$ is the (i,j)-th element of v-th view similarity matrix $\mathbf{A}_x^{(v)}$. Afterward, we calculate the spectral embedding $\mathbf{H}^{(v)}$ of $\mathbf{A}_x^{(v)}$, and use $\mathbf{H}^{(v)}$ to construct similarity matrix $\mathbf{A}_h^{(v)}$ via Eq. (29).

4.4.2 Initialization of $\mathbf{H}^*$ and $\mathbf{Y}$

In our experiment, we concatenate the features of multi-view data firstly. Then, we construct a similarity matrix $\mathbf{S}_c$ via adaptive neighbor graph learning. The $\mathbf{H}^*$ is initialized as the spectral embedding of $\mathbf{S}_c$. Finally, we perform k-means clustering on $\mathbf{H}^*$ to obtain discrete clustering partitions which are used to initialize matrix $\mathbf{Y}$.

4.4.3 Computational Complexity

The main complexity of the proposed method lies in solving $\mathbf{Y}$, $\mathbf{H}^*$. For updating $\mathbf{Y}$, it costs $\mathcal{O}(nk)$ in each iteration, where n and k are the number of samples and clusters, respectively. For updating $\mathbf{H}^*$, it costs $\mathcal{O}(nk^2)$ to solve singlur value decomposion in each iteration. For computing V views similarity graphs $\{\mathbf{A}_x^{(v)}\}_{v=1}^{V}$ and embedding matrices $\{\mathbf{H}^{(v)}\}_{v=1}^{V}$, it cost $\mathcal{O}(Vn^2)$ and $\mathcal{O}(Vn^2k)$, respectively. Therefore, the total complexity of Algorithm 1 is $\mathcal{O}(nkt+nk^2t+Vn^2+Vn^2k)$, where t represents the number of iterations.

4.4.4 Convergence Property

Theoretically proving the whole convergence of the method is not easy. Fortunately, each sub-problem can obtain a locally optimal solution in each iteration and convergence well. At the same time, the proposed method is upper bounded by the sum of the k largest eigenvalues of the unified graph $\mathbf{A}^*$. Therefore, the proposed method convergences. Besides, the proposed method converges well in reality, which will be verified in the next section.

TABLE 1
Clustering performance on six multi-view datasets. The highest and the second highest values under each metric are **bolded** and underlined, respectively.

Datasets	Methods	F-score	Precision	Recall	NMI	AR	ACC	Purity
MSRV1	CoReg	0.7502±0.0083	0.7467±0.0111	0.7538±0.0061	0.7491±0.0055	0.7098±0.0099	0.8533±0.0078	0.8533±0.0078
	AASC	0.6918±0.0049	0.6605±0.0053	0.7262±0.0045	0.7214±0.0036	0.6394±0.0058	0.7586±0.0027	0.8062±0.0027
	RMSC	0.6881±0.0057	0.6636±0.0048	0.7144±0.0067	0.7108±0.0003	0.6357±0.0066	0.7712±0.0011	0.8062±0.0035
	MVGL	0.7405±0.0000	0.6856±0.0000	0.8049±0.0000	0.7832±0.0000	0.6947±0.0000	0.8190±0.0000	0.8190±0.0000
	WMSC	0.7026±0.0005	0.6810±0.0011	0.7257±0.0003	0.7224±0.0004	0.6529±0.0006	0.7888±0.0023	0.8095±0.0000
	AWP	0.7937±0.0000	0.7675±0.0000	0.8217±0.0000	0.7953±0.0000	0.7591±0.0000	0.8810±0.0000	0.8810±0.0000
	MCGC	0.7068±0.0000	0.6660±0.0000	0.7530±0.0000	0.7350±0.0000	0.6562±0.0000	0.7857±0.0000	0.8048±0.0000
	GMC	0.7240±0.0000	0.6672±0.0000	0.7915±0.0000	0.7733±0.0000	0.6751±0.0000	0.7571±0.0000	0.8095±0.0000
	SMSC	0.7373±0.0000	0.7292±0.0000	0.7455±0.0000	0.7466±0.0000	0.6944±0.0000	0.8333±0.0000	0.8333±0.0000
	CGD	0.6827±0.0002	0.6586±0.0008	0.7085±0.0008	0.7068±0.0004	0.6293±0.0002	0.7710±0.0021	0.8000±0.0000
	OPMC	0.7563±0.0000	0.7441±0.0000	0.7688±0.0000	0.7599±0.0000	0.7162±0.0000	0.8714±0.0000	0.8714±0.0000
	CDMGC	0.7127±0.0000	0.6554±0.0000	0.7931±0.0000	0.7622±0.0000	0.6609±0.0000	0.7810±0.0000	0.8048±0.0000
	Ours	**0.8252±0.0000**	**0.8189±0.0000**	**0.8315±0.0000**	**0.8226±0.0000**	**0.7967±0.0000**	**0.9095±0.0000**	**0.9095±0.0000**
LGG	CoReg	0.7471±0.0000	0.7663±0.0000	0.7288±0.0000	0.6462±0.0000	0.5968±0.0000	0.8614±0.0000	0.8614±0.0000
	AASC	0.8857±0.0000	0.8627±0.0000	0.9100±0.0000	0.7892±0.0000	0.8119±0.0000	0.9401±0.0000	0.9401±0.0000
	RMSC	0.9040±0.0000	0.8919±0.0000	0.9164±0.0000	0.8207±0.0000	0.8433±0.0000	0.9513±0.0000	0.9513±0.0000
	MVGL	0.8150±0.0000	0.7630±0.0000	0.8746±0.0000	0.7148±0.0000	0.6873±0.0000	0.8951±0.0000	0.8951±0.0000
	WMSC	0.8779±0.0000	0.9073±0.0000	0.8504±0.0000	0.7823±0.0000	0.8063±0.0000	0.9363±0.0000	0.9363±0.0000
	AWP	0.4954±0.0000	0.5018±0.0000	0.4891±0.0000	0.2037±0.0000	0.1896±0.0000	0.5880±0.0000	0.5993±0.0000
	MCGC	0.7027±0.0000	0.6368±0.0000	**0.9841±0.0000**	0.6073±0.0000	0.4181±0.0000	0.7640±0.0000	0.7640±0.0000
	GMC	0.8418±0.0000	0.7994±0.0000	0.8890±0.0000	0.7379±0.0000	0.7352±0.0000	0.9139±0.0000	0.9139±0.0000
	SMSC	0.8315±0.0000	0.8683±0.0000	0.7977±0.0000	0.6916±0.0000	0.7342±0.0000	0.8951±0.0000	0.8951±0.0000
	CGD	0.8690±0.0000	0.8961±0.0000	0.8435±0.0000	0.7780±0.0000	0.7918±0.0000	0.9326±0.0000	0.9326±0.0000
	OPMC	0.7625±0.0000	0.7412±0.0000	0.7852±0.0000	0.6087±0.0000	0.6086±0.0000	0.8614±0.0000	0.8614±0.0000
	CDMGC	0.8313±0.0000	0.8220±0.0000	0.8650±0.0000	0.7015±0.0000	0.7250±0.0000	0.8989±0.0000	0.8989±0.0000
	Ours	**0.9270±0.0000**	**0.9387±0.0000**	0.9207±0.0000	**0.8666±0.0000**	**0.8823±0.0000**	**0.9625±0.0000**	**0.9625±0.0000**
100leaves	CoReg	0.9347±0.0103	0.9023±0.0146	0.9710±0.0054	0.9861±0.0024	0.9341±0.0104	0.9387±0.0133	0.9509±0.0077
	AASC	0.8024±0.0661	0.6989±0.0938	0.9506±0.0083	0.9730±0.0054	0.8002±0.0670	0.8999±0.0186	0.9228±0.0133
	RMSC	0.9459±0.0127	0.9249±0.0219	0.9681±0.0043	0.9867±0.0018	0.9453±0.0128	0.9588±0.0080	0.9672±0.0059
	MVGL	0.9573±0.0000	0.9464±0.0000	0.9685±0.0000	0.9879±0.0000	0.9569±0.0000	0.9706±0.0000	0.9744±0.0000
	WMSC	0.9505±0.0061	0.9250±0.0100	0.9775±0.0038	0.9895±0.0011	0.9500±0.0062	0.9556±0.0065	0.9634±0.0047
	AWP	0.8582±0.0000	0.7727±0.0000	0.9650±0.0000	0.9707±0.0000	0.8567±0.0000	0.8625±0.0000	0.8669±0.0000
	MCGC	0.9589±0.0000	0.9502±0.0000	0.9677±0.0000	0.9872±0.0000	0.9585±0.0000	0.9725±0.0000	0.9763±0.0000
	GMC	0.9755±0.0000	0.9660±0.0000	0.9852±0.0000	0.9945±0.0000	0.9753±0.0000	0.9788±0.0000	0.9838±0.0000
	SMSC	0.9743±0.0000	0.9696±0.0000	0.9790±0.0000	0.9916±0.0000	0.9740±0.0000	0.9812±0.0000	0.9856±0.0000
	CGD	0.9533±0.0088	0.9354±0.0140	0.9720±0.0045	0.9885±0.0018	0.9528±0.0089	0.9627±0.0083	0.9690±0.0068
	OPMC	0.6736±0.0000	0.5387±0.0000	0.8988±0.0000	0.9270±0.0000	0.6698±0.0000	0.7106±0.0000	0.7331±0.0000
	CDMGC	0.9816±0.0000	0.9738±0.0000	**0.9897±0.0000**	0.9957±0.0000	0.9815±0.0000	0.9825±0.0000	0.9875±0.0000
	Ours	**0.9886±0.0000**	**0.9880±0.0000**	0.9892±0.0000	**0.9960±0.0000**	**0.9885±0.0000**	**0.9944±0.0000**	**0.9944±0.0000**
ORL	CoReg	0.7236±0.0206	0.6731±0.0288	0.7828±0.0142	0.9033±0.0063	0.7167±0.0212	0.7933±0.0162	0.8241±0.0106
	AASC	0.5862±0.0399	0.4875±0.0498	0.7385±0.0241	0.8704±0.0121	0.5746±0.0414	0.7161±0.0245	0.7707±0.0200
	RMSC	0.7312±0.0226	0.7062±0.0189	0.7774±0.0200	0.8997±0.0071	0.7247±0.0232	0.8075±0.0194	0.8411±0.0133
	MVGL	0.3023±0.0000	0.1883±0.0000	0.7661±0.0000	0.8295±0.0000	0.2761±0.0000	0.6325±0.0000	0.6975±0.0000
	WMSC	0.7037±0.0215	0.6490±0.0286	0.7690±0.0161	0.8968±0.0067	0.6963±0.0222	0.7909±0.0135	0.8227±0.0127
	AWP	0.6312±0.0000	0.5283±0.0000	0.7839±0.0000	0.8728±0.0000	0.6210±0.0000	0.7050±0.0000	0.7125±0.0000
	MCGC	0.5319±0.0002	0.4507±0.0003	0.7478±0.0000	0.8346±0.0000	0.5191±0.0002	0.6950±0.0000	0.7300±0.0000
	GMC	0.3840±0.0000	0.2519±0.0000	0.8078±0.0000	0.8628±0.0000	0.3621±0.0000	0.6400±0.0000	0.7225±0.0000
	SMSC	0.7006±0.0000	0.6333±0.0000	0.7839±0.0000	0.8998±0.0000	0.6929±0.0000	0.7750±0.0000	0.8175±0.0000
	CGD	0.6962±0.0178	0.6372±0.0242	0.7676±0.0140	0.8947±0.0058	0.6885±0.0184	0.7690±0.0161	0.8099±0.0137
	OPMC	0.4802±0.0000	0.4169±0.0000	0.5661±0.0000	0.7948±0.0000	0.4663±0.0000	0.6225±0.0000	0.6575±0.0000
	CDMGC	0.3401±0.0000	0.2160±0.0000	0.8106±0.0000	0.8530±0.0000	0.3159±0.0000	0.6425±0.0000	0.7250±0.0000
	Ours	**0.7984±0.0000**	**0.7677±0.0000**	**0.8317±0.0000**	**0.9259±0.0000**	**0.7936±0.0000**	**0.8575±0.0000**	**0.8800±0.0000**
handwritten	CoReg	0.9606±0.0000	**0.9603±0.0000**	0.9609±0.0000	0.9535±0.0000	0.9562±0.0000	0.9800±0.0000	0.9800±0.0000
	AASC	0.8507±0.0002	0.8382±0.0001	0.8635±0.0004	0.8838±0.0002	0.8339±0.0002	0.8770±0.0003	0.8770±0.0003
	RMSC	0.3751±0.0196	0.2466±0.0177	0.7865±0.0052	0.6449±0.0177	0.2634±0.0254	0.4589±0.0135	0.5503±0.0220
	MVGL	0.8507±0.0000	0.7927±0.0000	0.9179±0.0000	0.9080±0.0000	0.8328±0.0000	0.8520±0.0000	0.8820±0.0000
	WMSC	0.8126±0.0003	0.8031±0.0003	0.8222±0.0003	0.8565±0.0003	0.7916±0.0003	0.8460±0.0002	0.8460±0.0002
	AWP	0.9550±0.0000	0.9548±0.0000	0.9552±0.0000	0.9472±0.0000	0.9500±0.0000	0.9770±0.0000	0.9770±0.0000
	MCGC	0.9477±0.0000	0.9473±0.0000	0.9480±0.0000	0.9362±0.0000	0.9419±0.0000	0.9735±0.0000	0.9735±0.0000
	GMC	0.8715±0.0000	0.8308±0.0000	0.9164±0.0000	0.9110±0.0000	0.8565±0.0000	0.8840±0.0000	0.8840±0.0000
	SMSC	0.8474±0.0000	0.8347±0.0000	0.8605±0.0000	0.8802±0.0000	0.8302±0.0000	0.8715±0.0000	0.8715±0.0000
	CGD	0.8204±0.0000	0.7756±0.0000	0.8708±0.0000	0.8664±0.0000	0.7993±0.0000	0.8210±0.0000	0.8640±0.0000
	OPMC	0.7566±0.0000	0.7180±0.0000	0.7995±0.0000	0.7913±0.0000	0.7280±0.0000	0.8045±0.0000	0.8110±0.0000
	CDMGC	0.8563±0.0000	0.7978±0.0000	0.9244±0.0000	0.9168±0.0000	0.8391±0.0000	0.8540±0.0000	0.8850±0.0000
	Ours	**0.9610±0.0000**	0.9602±0.0000	**0.9619±0.0000**	**0.9558±0.0000**	**0.9567±0.0000**	**0.9805±0.0000**	**0.9805±0.0000**
scene-15	CoReg	0.3292±0.0041	0.3168±0.0051	0.3433±0.0049	0.4643±0.0050	0.2773±0.0045	0.4409±0.0115	0.4910±0.0090
	AASC	0.3380±0.0045	0.2839±0.0057	0.4175±0.0039	0.4786±0.0030	0.2786±0.0053	0.4544±0.0118	0.4976±0.0129
	RMSC	0.3305±0.0038	0.3248±0.0032	0.6447±0.0451	0.4636±0.0056	0.2799±0.0040	0.4459±0.0051	0.4966±0.0066
	MVGL	0.2104±0.0000	0.1246±0.0000	0.6762±0.0000	0.3658±0.0000	0.1060±0.0000	0.2849±0.0000	0.2981±0.0000
	WMSC	0.3517±0.0036	0.3317±0.0032	0.3743±0.0070	0.5001±0.0024	0.3004±0.0037	0.4520±0.0072	0.5298±0.0092
	AWP	0.3239±0.0000	0.2960±0.0000	0.3577±0.0000	0.4545±0.0000	0.2686±0.0000	0.4334±0.0000	0.4821±0.0000
	MCGC	0.2955±0.0000	0.2438±0.0000	0.7163±0.0000	0.4355±0.0000	0.2311±0.0000	0.3987±0.0000	0.4417±0.0000
	GMC	0.3055±0.0000	0.2061±0.0000	0.5901±0.0000	0.5079±0.0000	0.2262±0.0000	0.3837±0.0000	0.4107±0.0000
	SMSC	0.3249±0.0000	0.2929±0.0000	0.3648±0.0000	0.4693±0.0000	0.2689±0.0000	0.4163±0.0000	0.4742±0.0000
	CGD	0.3579±0.0043	**0.3443±0.0046**	0.3726±0.0039	0.4996±0.0032	0.3082±0.0047	0.4732±0.0034	0.5228±0.0038
	OPMC	0.2811±0.0000	0.2750±0.0000	0.2874±0.0000	0.4094±0.0000	0.2265±0.0000	0.3871±0.0000	0.4421±0.0000
	CDMGC	0.2475±0.0000	0.1480±0.0000	**0.8363±0.0000**	0.4460±0.0000	0.1492±0.0000	0.3104±0.0000	0.3195±0.0000
	Ours	**0.3690±0.0000**	0.3315±0.0000	0.4160±0.0000	**0.5254±0.0000**	**0.3164±0.0000**	**0.4845±0.0000**	**0.5298±0.0000**

TABLE 2
The clustering results in terms of ACC and average CPU running time (sec.) of different methods on six datasets.

Method	MSRCV1		LGG		100leaves		ORL		handwritten		sence-15	
	ACC	running time	ACC	running time	ACC	running time	ACC	running time	ACC	running time	ACC	running time
CoReg	0.8533	1.0231	0.8614	0.4716	0.9387	41.4690	0.7933	3.8981	0.9800	29.4093	0.4409	154.6188
AASC	0.7586	2.4697	0.9401	0.4268	0.8999	13.0357	0.7161	2.6324	0.8770	3.7248	0.4544	18.1562
RMSC	0.7712	0.9079	0.9513	0.9331	0.9588	54.8059	0.8075	2.9887	0.4589	96.3689	0.4459	912.4662
MVGL	0.8190	3.0300	0.8951	2.6115	0.9706	79.5868	0.6325	8.3227	0.8520	63.8117	0.2849	1096.8229
WMSC	0.7888	0.9545	0.9363	0.3607	0.9556	13.8862	0.7909	1.8781	0.8460	5.9886	0.4520	21.4511
AWP	0.8810	0.2905	0.5880	0.1106	0.8625	2.5414	0.7050	0.4979	0.9770	5.7354	0.4334	13.1430
MCGC	0.7857	1.7310	0.7640	1.3868	0.9725	49.2979	0.6950	5.6132	0.9735	23.5376	0.3987	160.3354
GMC	0.7571	0.9557	0.9139	0.4879	0.9788	5.3476	0.6400	0.5898	0.8840	25.2614	0.3837	312.6088
SMSC	0.8333	0.5524	0.8951	0.2816	0.9812	10.7882	0.7750	0.9497	0.8715	12.5581	0.4163	38.4127
CGD	0.7710	3.3349	0.9326	3.7772	0.9627	119.3134	0.7690	11.0049	0.8210	333.5162	0.4732	1710.3768
OPMC	0.8714	0.1976	0.8614	0.4816	0.7106	0.6805	0.6225	2.9695	0.8045	0.8778	0.3871	1.0682
CDMGC	0.7810	0.1279	0.8989	0.1840	0.9825	4.2940	0.6425	0.4106	0.8540	15.9718	0.3104	136.6304
Ours	0.9095	0.3395	0.9625	0.8105	0.9944	10.1792	0.8575	0.8514	0.9805	19.0913	0.4845	100.2653

Algorithm 1: Unified One-step Multi-view Spectral Clustering

Input: Multi-view data $\{\mathbf{X}^{(v)}\}_{v=1}^{V}$, parameter λ, clustering number k;
Output: Clustering results $\mathbf{Y}$;

Compute similarity graphs $\{\mathbf{A}_x^{(v)}\}_{v=1}^{V}$ and $\{\mathbf{A}_h^{(v)}\}_{v=1}^{V}$;
Initialize $t = 1$, $\mathbf{Y}$, $\mathbf{H}^*$, $\epsilon = 10^{-6}$;
while *not converged* **do**
  Update variable $\boldsymbol{\alpha}$ via Eq. (26);
  Update variable $\boldsymbol{\beta}$ via Eq. (28);
  Update Unified graph $\mathbf{A}^*$ via Eq. (15);
  Update variable $\mathbf{H}^*$ via Eq. (24);
  Update variable $\mathbf{Y}$ via Eq. (21);
  Check convergence condition $t \geq 2$ and $|\frac{obj(t-1)-obj(t)}{obj(t)}| < \epsilon$;
  $t = t + 1$;
end
return clustering results $\mathbf{Y}$;

5 Experiments

5.1 Datasets

We evaluate the proposed method on six widely used multi-view datasets. The details of the multi-view datasets are as follows,

MSRCV1[1]: It consists of 210 scene recognition images belonging to seven classes. Each image is represented by five different feature sets, i.e., 256 dimensions LBP, 100 dimensions HOG, 512 dimensions GIST, 1302 dimensions CENTRIST, and 210 dimensions SIFT.

LGG [44]: It is a cancer dataset that consists of 267 samples of 3 classes. For each sample, the 2000 dimension DNA methylation, 2000 dimensions gene expression, 333 dimensions microRNA expression, and 209 dimensions reverse-phase protein array expression features are extracted.

100leaves[2]: This set contains 1600 samples of 100 plant species. Shape descriptor, texture histogram, and fine-scale margin are extracted to depict each sample.

ORL[3]: It consists of 400 face recognition images from 40 different persons. For each image, three types of features, including, 4096 dimension intensity, 3304 dimension LBP, and 6750 dimension Gabor features are extracted.

handwritten[4]: It consists of 2000 handwritten digital images of 0 to 9. Each sample is represented in terms of six different feature sets, i.e., 76 dimensions FOU, 216 dimensions FAC, 64 dimensions KAR, 240 dimensions Pix, 47 dimensions ZER, and six dimensions MOR.

sence-15 [45]: It consists of 4485 scene images of 15 classes with both indoor and outdoor environments. For each image, the 20 dimensions PHOG, 59 dimensions GIST, and 40 dimensions LBP features are adopted.

5.2 Compared Methods

To validate the superiority of our method, we compare it with 12 methods. These methods are listed as follows:

CoReg [34]: Co-regularized multi-view spectral clustering.

AASC [46]: Affinity aggregation for spectral clustering.

RMSC [47]: Robust multi-view spectral clustering via low-rank and sparse decomposition.

MVGL [48]: Graph Learning for Multiview Clustering.

WMSC [49]: Weighted multi-view spectral clustering based on spectral perturbation.

AWP [50]: Multiview Clustering via Adaptively Weighted Procrustes.

MCGC [51]: Multiview consensus graph clustering.

GMC [24]: Graph-based multi-view clustering.

SMSC [52]: Multi-view spectral clustering via integrating nonnegative embedding and spectral embedding.

CGD [53]: Multi-View Clustering via Cross-View Graph Diffusion.

OPMC [54]: One-pass Multi-view Clustering for Large-scale Data.

CDMGC [55]: Measuring Diversity in Graph Learning: A Unified Framework for Structured Multi-view Clustering.

5.3 Experimental Settings

In our experiments, we firstly scale each feature of the multi-view data into range $[0, 1]$, then generate multiple graphs for all methods via the adaptive neighbor graph learning method. The number of nearest neighbors of the adaptive neighbor graph learning method and that of other compared methods are set to 15 for all datasets. For the proposed

1. https://www.microsoft.com/en-us/research/project/
2. https://archive.ics.uci.edu/ml/datasets/One-hundred+plant+species+leaves+data+set
3. http://www.uk.research.att.com/facedatabase. html
4. https://archive.ics.uci.edu/ml/datasets/Multiple+Features

method, we search parameter λ in rang $[0 : 0.05 : 1]$ with a grid search scheme. The parameters of the compared methods are tuned in the ranges as suggestions in their papers. For each compared method, we repeat each experiment 20 times to suppress the impacts of the random initialization in k-means and report average results and standard deviations. In addition, seven evaluation metrics, including accuracy (ACC), normalized mutual information (NMI), adjusted Rand index (ARI), F-score, Precision, Recall, and Purity are employed to evaluate the performance. Note that higher values of these metrics indicate the better performance of clustering results. All methods are implemented with MATLAB R2020a and conducted on a PC with Ubuntu 20.04 system, Inter Core i7-7700 3.6GHz CPU, and 32GB RAM.

5.4 Performance Evaluation

We report the clustering results measured by seven metrics on the six multi-view datasets in TABLE 1. From this table, we obtain the following observations:

- The clustering performance of the proposed method consistently outperforms that of all compared methods. For example, the proposed method captures a better performance of about 1.13 and 0.8 percentage in terms of ACC and NMI than the second performer (CGD) on the scene-15 dataset. Therefore, it demonstrates the effectiveness of proposed method which jointly explores the information of the similarity graphs and spectral embedding matrices and captures clustering indicators in a one-step manner.
- The proposed method outperforms SMSC which carries out matrix factorization on multiple Laplacian matrices to obtain a consensus clustering indicator matrix for clustering. Since SMSC ignores the information of view-specific embedding matrices and only considers the non-negativity of clustering indicators, the clustering performance is limited.
- Among all competitors, our method, AWP, MCGC, GMC, SMSC, and CDMGC can obtain the clustering partition results without other post-processing. Furthermore, the GMC method jointly learns the view-specific graphs, view-weights, and a Laplacian rank constrained unified graph, from which the clustering results can be directly captured. And GMC method usually gains promising clustering performance in some datasets.

To further compare the proposed method with competitors, we record the average CPU running time as well as the ACC performance of each method in TABLE 2. From this table, we obtain the following observations:

- The proposed method can achieve the best clustering performance at a considerable running time cost compared with other competitors.
- Most competitors, such as, CoReg, RMSC, MVGL, and MCGC, etc, need to perform singular value decomposition on the Laplacian matrix with size $n \times n$, which is very time-consuming.
- CGD method carries out the cross-view graph diffusion to capture the complementary information among diverse views. however, the computing time is intolerable when the number of samples is extensive large.

Totally, the proposed method is efficient and effective for clustering multi-view data.

5.5 Ablation Studies

The proposed method consists of two key components, i.e., learning unified graph and one-step manner. To verify the effectiveness of these two parts, we formulated three methods for comparison, including UOMvSC-wo-O, UOMvSC-wo-Ah, and UOMvSC-wo-Ax. For UOMvSC-wo-O method, we first obtain spectral embedding matrix from the learned unified graph, then perform k-means to capture the discrete clustering results. For UOMvSC-wo-Ah and and UOMvSC-wo-Ax methods, we only utilize view-specific graphs or spectral embedding matrices to perform clustering. The clustering performance measured by seven metrics on six benchmark datasets are reported in TABLE 3. As seen from results, the clustering performance decreases when the key components of the proposed method are dropped out. This strongly indicates the effectiveness of the proposed components.

5.6 Parameter Sensitivity and Convergence Analysis

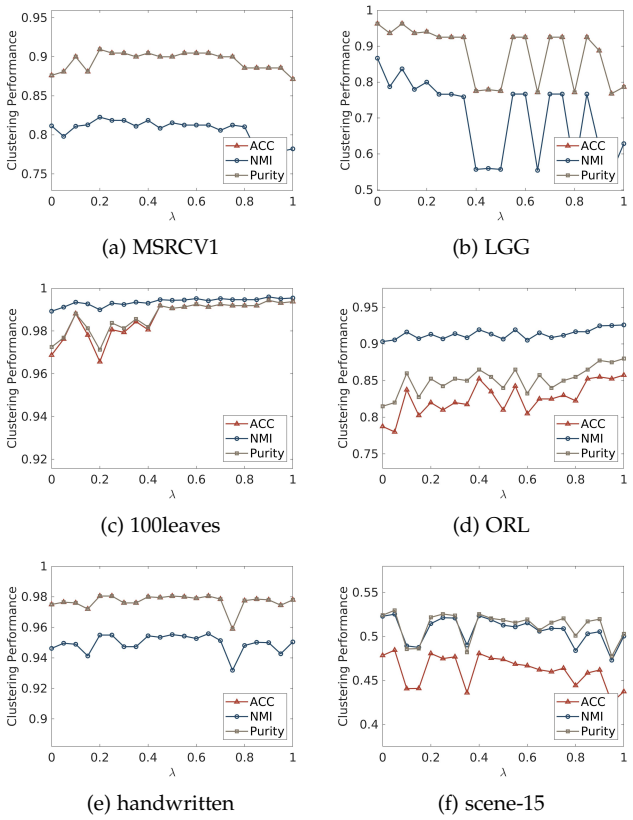
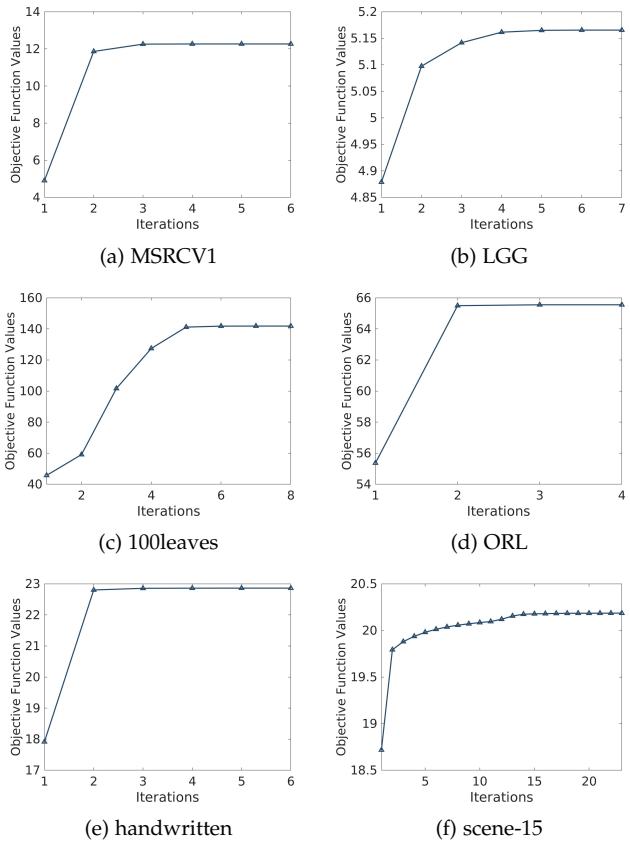


Fig. 2. The parameter sensitivity six multi-view datasets in terms of ACC, NMI, and Purity, respectively.

The proposed method contains a balance parameter in Eq.(16), i.e., λ, which is utilized to negotiate the information of graph and embedding matrix. To study the parameter sensitivity, we conduct experiments on six multi-view

TABLE 3
The ablation study of the proposed method on six benchmark multi-view datasets. The highest and the second highest values under each metric are **bolded** and underlined, respectively.

Datasets	Methods	F-score	Precision	Recall	NMI	AR	ACC	Purity
MSRV1	UOMvSC-wo-O	0.7017±0.0000	0.6798±0.0000	0.7251±0.0000	0.7219±0.0000	0.6519±0.0000	0.7826±0.0023	0.8095±0.0000
	UOMvSC-wo-Ah	0.7705±0.0000	0.7597±0.0000	0.7816±0.0000	0.7821±0.0000	0.7329±0.0000	0.8714±0.0000	0.8714±0.0000
	UOMvSC-wo-Ax	0.7826±0.0000	0.7668±0.0000	0.7990±0.0000	0.8115±0.0000	0.7467±0.0000	0.8762±0.0000	0.8762±0.0000
	UOMvSC	**0.8252±0.0000**	**0.8189±0.0000**	**0.8315±0.0000**	**0.8226±0.0000**	**0.7967±0.0000**	**0.9095±0.0000**	**0.9095±0.0000**
LGG	UOMvSC-wo-O	0.8804±0.0000	0.9102±0.0000	0.8524±0.0000	0.7774±0.0000	0.8102±0.0000	0.9363±0.0000	0.9363±0.0000
	UOMvSC-wo-Ah	0.6632±0.0000	0.6579±0.0000	0.6685±0.0000	0.6287±0.0000	0.4521±0.0000	0.7865±0.0000	0.7865±0.0000
	UOMvSC-wo-Ax	0.9231±0.0000	**0.9387±0.0000**	0.9080±0.0000	**0.8666±0.0000**	0.8768±0.0000	**0.9625±0.0000**	**0.9625±0.0000**
	UOMvSC	**0.9270±0.0000**	**0.9387±0.0000**	**0.9207±0.0000**	**0.8666±0.0000**	**0.8823±0.0000**	**0.9625±0.0000**	**0.9625±0.0000**
100leaves	UOMvSC-wo-O	0.9615±0.0064	0.9439±0.0088	0.9815±0.0032	0.9912±0.0016	0.9612±0.0065	0.9668±0.0059	0.9726±0.0045
	UOMvSC-wo-Ah	0.9873±0.0000	0.9867±0.0000	0.9880±0.0000	0.9954±0.0000	0.9872±0.0000	0.9938±0.0000	0.9938±0.0000
	UOMvSC-wo-Ax	0.9556±0.0000	0.9405±0.0000	0.9712±0.0000	0.9893±0.0000	0.9552±0.0000	0.9688±0.0000	0.9725±0.0000
	UOMvSC	**0.9886±0.0000**	**0.9880±0.0000**	**0.9892±0.0000**	**0.9960±0.0000**	**0.9885±0.0000**	**0.9944±0.0000**	**0.9944±0.0000**
ORL	UOMvSC-wo-O	0.7704±0.0165	0.7420±0.0209	0.8037±0.0153	0.9171±0.0065	0.7649±0.0169	0.8320±0.0135	0.8550±0.0125
	UOMvSC-wo-Ah	**0.7984±0.0000**	**0.7677±0.0000**	**0.8317±0.0000**	**0.9259±0.0000**	**0.7936±0.0000**	**0.8575±0.0000**	**0.8800±0.0000**
	UOMvSC-wo-Ax	0.7260±0.0000	0.6772±0.0000	0.7822±0.0000	0.9032±0.0000	0.7192±0.0000	0.7875±0.0000	0.8150±0.0000
	UOMvSC	**0.7984±0.0000**	**0.7677±0.0000**	**0.8317±0.0000**	**0.9259±0.0000**	**0.7936±0.0000**	**0.8575±0.0000**	**0.8800±0.0000**
handwritten	UOMvSC-wo-O	0.8643±0.0000	0.8599±0.0000	0.9202±0.0000	0.9055±0.0000	0.8492±0.0000	0.8845±0.0000	0.8845±0.0000
	UOMvSC-wo-Ah	0.9561±0.0000	0.9551±0.0000	0.9570±0.0000	0.9505±0.0000	0.9512±0.0000	0.9780±0.0000	0.9780±0.0000
	UOMvSC-wo-Ax	0.9502±0.0000	0.9490±0.0000	0.9514±0.0000	0.9462±0.0000	0.9447±0.0000	0.9750±0.0000	0.9750±0.0000
	UOMvSC	**0.9610±0.0000**	**0.9602±0.0000**	**0.9619±0.0000**	**0.9558±0.0000**	**0.9567±0.0000**	**0.9805±0.0000**	**0.9805±0.0000**
scene-15	UOMvSC-wo-O	0.3518±0.0060	**0.3357±0.0011**	0.3697±0.0121	0.4966±0.0014	0.3012±0.0057	0.4626±0.0046	0.5268±0.0009
	UOMvSC-wo-Ah	**0.3690±0.0000**	0.3315±0.0000	**0.4160±0.0000**	0.5004±0.0000	**0.3164±0.0000**	0.4375±0.0000	0.5030±0.0000
	UOMvSC-wo-Ax	0.3567±0.0000	0.3231±0.0000	0.3982±0.0000	0.5230±0.0000	0.3036±0.0000	0.4785±0.0000	0.5242±0.0000
	UOMvSC	**0.3690±0.0000**	0.3315±0.0000	**0.4160±0.0000**	**0.5254±0.0000**	**0.3164±0.0000**	**0.4845±0.0000**	**0.5298±0.0000**



(a) MSRCV1 (b) LGG (c) 100leaves (d) ORL (e) handwritten (f) scene-15

Fig. 3. Convergence curves of the proposed method on six multi-view datasets.

datasets. We report the clustering performance measured by ACC, NMI, and Purity varying with different λ in Fig. 2. As can be seen, the clustering performance of the proposed method slightly fluctuates with different λ. When λ is 0 or 1, the unified graph $\mathbf{A}^*$ in Eq. (15) is respectively degraded into the original graph and inner product of spectral embedding matrix, and clustering performance is reduced in such cases. It indicates the effectiveness of jointly exploring the information of graph and embedding matrix.

Fig. 3 gives the objective function values of the proposed method on six multi-view datasets varying with different iterations. From this figure, we observe that the objective function values of the proposed method increase very fast and converge within several iteration times. Thus, the proposed method contains strong convergence.

6 Conclusions

In this paper, we proposed a novel multi-view spectral clustering method by jointly utilizing the information of view-specific graphs and embedding matrices. A unified graph is introduced by combining view-specific graphs and embedding matrices. Based on the unified graph, we directly solve the spectral clustering problem to obtain the clustering indicator. An efficient optimization algorithm was designed to solve the resultant model. Extensive experiments on various benchmark datasets were conducted to verify the efficacy of our method.

The proposed method consists of three aspects of strengths. First, the proposed method can achieve superior clustering results compared with other methods. Second, it cost a considerable running time to partition data. Third, the motivation and formulation of the proposed are simple and intuitive, which makes it can be further extended to copy with other tasks, i.e., incomplete multi-view clustering. The proposed method contains a parameter λ which is used to balance $\mathbf{A}_x^{(v)}$ and $\mathbf{A}_h^{(v)}$. Thus, it needs to carefully adjust this parameter to obtain considerable results for real-world applications. In this paper, we mainly focus on the approach to obtain a more accurate discrete partition matrix from the similarity graphs for clustering. Thus, we simply utilize the pre-computed similarity graphs to perform clustering. Also, adaptively learning the similarity graphs can provide better performance. In our future work, we propose to integrate adaptive graph learning and one-step spectral clustering into a unified framework.

ACKNOWLEDGMENTS

The authors wish to gratefully acknowledge the anonymous reviewers for the constructive comments of this paper.

REFERENCES

[1] I. S. Dhillon, "Co-clustering documents and words using bipartite spectral graph partitioning," in *International Conference on Knowledge Discovery and Data Mining*, 2001, pp. 269–274.

[2] X. Dong, L. Liu, L. Zhu, Z. Cheng, and H. Zhang, "Unsupervised deep k-means hashing for efficient image retrieval and clustering," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 8, pp. 3266–3277, 2021.

[3] N. Veldt, D. F. Gleich, and A. Wirth, "A correlation clustering framework for community detection," in *World Wide Web Conference*, 2018, pp. 439–448.

[4] X. Zhao, Q. Dai, J. Wu, H. Peng, M. Liu, X. Bai, J. Tan, S. Wang, and P. Yu, "Multi-view tensor graph neural networks through reinforced aggregation," *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[5] V. Y. Kiselev, K. Kirschner, M. T. Schaub, T. Andrews, A. Yiu, T. Chandra, K. N. Natarajan, W. Reik, M. Barahona, A. R. Green *et al.*, "Sc3: consensus clustering of single-cell rna-seq data," *Nature Methods*, vol. 14, no. 5, pp. 483–486, 2017.

[6] P. Zhou, L. Du, L. Shi, H. Wang, and Y.-D. Shen, "Recovery of corrupted multiple kernels for clustering," in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

[7] C. Tang, X. Liu, X. Zhu, J. Xiong, M. Li, J. Xia, X. Wang, and L. Wang, "Feature selective projection with low-rank embedding and dual laplacian regularization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 9, pp. 1747–1760, 2019.

[8] 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.

[9] Z. Zhang, L. Liu, Y. Luo, Z. Huang, F. Shen, H. T. Shen, and G. Lu, "Inductive structure consistent hashing via flexible semantic calibration," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 10, pp. 4514–4528, 2020.

[10] K. Yi, J. Wu, C. Gan, A. Torralba, P. Kohli, and J. B. Tenenbaum, "Neural-symbolic vqa: disentangling reasoning from vision and language understanding," in *International Conference on Neural Information Processing Systems*, 2018, pp. 1039–1050.

[11] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[12] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1. IEEE, 2005, pp. 886–893.

[13] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[14] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[15] C. Tang, L. Xinwang, X. Zheng, W. Li, J. Xiong, L. Wang, A. Zomaya, and A. Longo, "Defusionnet: Defocus blur detection via recurrently fusing and refining discriminative multi-scale deep features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 2, pp. 955–968, 2022.

[16] Z. Zhang, H. Luo, L. Zhu, G. Lu, and H. T. Shen, "Modality-invariant asymmetric networks for cross-modal hashing," *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[17] Y. Wang, L. Wu, X. Lin, and J. Gao, "Multiview spectral clustering via structured low-rank matrix factorization," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 10, pp. 4833–4843, 2018.

[18] Z. Zhang, L. Liu, F. Shen, H. T. Shen, and L. Shao, "Binary multi-view clustering," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 7, pp. 1774–1782, 2018.

[19] Z. Yang, N. Liang, W. Yan, Z. Li, and S. Xie, "Uniform distribution non-negative matrix factorization for multiview clustering," *IEEE Transactions on Cybernetics*, vol. 51, no. 6, pp. 3249–3262, 2020.

[20] H. Zhao, Z. Ding, and Y. Fu, "Multi-view clustering via deep matrix factorization," in *AAAI Conference on Artificial Intelligence*, 2017, pp. 2921–2927.

[21] 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.

[22] Z. Kang, W. Zhou, Z. Zhao, J. Shao, M. Han, and Z. Xu, "Large-scale multi-view subspace clustering in linear time," in *AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, 2020, pp. 4412–4419.

[23] P. Zhang, X. Liu, J. Xiong, S. Zhou, W. Zhao, E. Zhu, and Z. Cai, "Consensus one-step multi-view subspace clustering," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2020.

[24] H. Wang, Y. Yang, and B. Liu, "Gmc: Graph-based multi-view clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 6, pp. 1116–1129, 2020.

[25] P. Zhou, Y.-D. Shen, L. Du, F. Ye, and X. Li, "Incremental multi-view spectral clustering," *Knowledge-Based Systems*, vol. 174, pp. 73–86, 2019.

[26] Z. Lin and Z. Kang, "Graph filter-based multi-view attributed graph clustering," in *International Joint Conference on Artificial Intelligence*, 2021, pp. 19–26.

[27] C.-D. Wang, J.-H. Lai, and P. S. Yu, "Multi-view clustering based on belief propagation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 4, pp. 1007–1021, 2016.

[28] Z. Lin, Z. Kang, L. Zhang, and L. Tian, "Multi-view attributed graph clustering," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2021.

[29] J. Xu, Y. Ren, G. Li, L. Pan, C. Zhu, and Z. Xu, "Deep embedded multi-view clustering with collaborative training," *Information Sciences*, vol. 573, pp. 279–290, 2021.

[30] Q. Wang, Z. Ding, Z. Tao, Q. Gao, and Y. Fu, "Partial multi-view clustering via consistent gan," in *International Conference on Data Mining*. IEEE, 2018, pp. 1290–1295.

[31] Y. Xie, B. Lin, Y. Qu, C. Li, W. Zhang, L. Ma, Y. Wen, and D. Tao, "Joint deep multi-view learning for image clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 11, pp. 3594–3606, 2021.

[32] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems*, 2002, pp. 849–856.

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

[34] A. Kumar, P. Rai, and H. Daume, "Co-regularized multi-view spectral clustering," *Advances in Neural Information Processing Systems*, vol. 24, pp. 1413–1421, 2011.

[35] J. Liu, C. Wang, J. Gao, and J. Han, "Multi-view clustering via joint nonnegative matrix factorization," in *SIAM International Conference on Data Mining*. SIAM, 2013, pp. 252–260.

[36] G. Trigeorgis, K. Bousmalis, S. Zafeiriou, and B. W. Schuller, "A deep matrix factorization method for learning attribute representations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 3, pp. 417–429, 2016.

[37] 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.

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

[39] Z. Li, C. Tang, X. Liu, X. Zheng, G. Yue, W. Zhang, and E. Zhu, "Consensus graph learning for multi-view clustering," *IEEE Transactions on Multimedia*, pp. 1–1, 2021.

[40] Z. Li, C. Tang, J. Chen, C. Wan, W. Yan, and X. Liu, "Diversity and consistency learning guided spectral embedding for multi-view clustering," *Neurocomputing*, vol. 370, pp. 128–139, 2019.

[41] C. Bauckhage, "K-means clustering is matrix factorization," *arXiv preprint arXiv:1512.07548*, 2015.

[42] X. Chen, W. Hong, F. Nie, D. He, M. Yang, and J. Z. Huang, "Spectral clustering of large-scale data by directly solving normalized cut," in *International Conference on Knowledge Discovery and Data Mining*, 2018, pp. 1206–1215.

[43] J. Huang, F. Nie, and H. Huang, "Spectral rotation versus k-means in spectral clustering," in *AAAI Conference on Artificial Intelligence*, 2013, pp. 431–437.

[44] C. G. A. R. Network, "Comprehensive, integrative genomic analysis of diffuse lower-grade gliomas," *New England Journal of Medicine*, vol. 372, no. 26, pp. 2481–2498, 2015.
[45] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2. IEEE, 2005, pp. 524–531.
[46] H.-C. Huang, Y.-Y. Chuang, and C.-S. Chen, "Affinity aggregation for spectral clustering," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 773–780.
[47] R. Xia, Y. Pan, L. Du, and J. Yin, "Robust multi-view spectral clustering via low-rank and sparse decomposition," in *AAAI Conference on Artificial Intelligence*, vol. 28, no. 1, 2014.
[48] K. Zhan, C. Zhang, J. Guan, and J. Wang, "Graph learning for multiview clustering," *IEEE Transactions on Cybernetics*, vol. 48, no. 10, pp. 2887–2895, 2017.
[49] L. Zong, X. Zhang, X. Liu, and H. Yu, "Weighted multi-view spectral clustering based on spectral perturbation," in *AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.
[50] F. Nie, L. Tian, and X. Li, "Multiview clustering via adaptively weighted procrustes," in *International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 2022–2030.
[51] K. Zhan, F. Nie, J. Wang, and Y. Yang, "Multiview consensus graph clustering," *IEEE Transactions on Image Processing*, vol. 28, no. 3, pp. 1261–1270, 2018.
[52] 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.
[53] 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 *AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, 2020, pp. 5924–5931.
[54] J. Liu, X. Liu, Y. Yang, L. Liu, S. Wang, W. Liang, and J. Shi, "One-pass multi-view clustering for large-scale data," in *International Conference on Computer Vision*, 2021, pp. 12 344–12 353.
[55] S. Huang, I. Tsang, Z. Xu, and J. C. Lv, "Measuring diversity in graph learning: A unified framework for structured multi-view clustering," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2021.

Chang Tang received his Ph.D. degree from Tianjin University, Tianjin, China in 2016. He joined the AMRL Lab of the University of Wollongong between Sep. 2014 and Sep. 2015. He is now a full professor at the School of Computer Science, China University of Geosciences, Wuhan, China. Dr. Tang has published 50+ peer-reviewed papers, including those in highly regarded journals and conferences such as IEEE T-PAMI, IEEE T-MM, IEEE T-KDE, IEEE T-HMS, ICCV, CVPR, IJCAI, AAAI and ACM MM, etc. He serves as an associate editor of BMC Bioinformatics, young editor of CAAI Transactions on Intelligence Technology, and Computer Engineering. He regularly serves on the Technical Program Committees or as Area Chair of some top conferences such as NIPS, ICML, CVPR, ICCV, ECCV, IJCAI, ICME and AAAI. His current research interests include multi-view learning and computer vision.

Zhenglai Li received the BE and MS degree from China University of Geosciences, Wuhan, China, in 2018 and 2021, respectively. Currently, he is pursuing a PhD degree in China University of Geosciences, Wuhan, China. His research interest focuses on multi-view learning.

JunWang received the B.E degree in Information Management and Information System from Hubei University of Technology, Wuhan, China, in 2019. He is currently working toward the Master's degree at the School of Computer Science, China University of Geosciences, Wuhan, China. His research interests include multi-view learning and computer vision.

Xinwang Liu received his PhD degree from National University of Defense Technology (NUDT), China. He is now Assistant Researcher of School of Computer Science, NUDT. His current research interests include kernel learning and unsupervised feature learning. Dr. Liu has published 60+ peer-reviewed papers, including those in highly regarded journals and conferences such as IEEE T-PAMI, IEEE T-IP, IEEE T-NNLS, ICCV, AAAI, IJCAI, etc. He regularly serves on the Technical Program Committees of top conferences such as NIPS, ICML, CVPR, ICCV, IJCAI and AAAI.

Wei Zhang received the B.E. degree from Zhejiang University in 2004, the M.S. degree from Liaoning University in 2008, and the Ph.D. degree from Shandong University of Science and Technology in 2018. He is currently an Associate Professor with the Shandong Computer Science Center (National Supercomputing Center in Jinan), Qilu University of Technology (Shandong Academy of Sciences). His research interests include future generation network architectures, edge computing and edge intelligence.

En Zhu received his PhD degree from National University of Defense Technology (NUDT), China. He is now Professor at School of Computer Science, NUDT, China. His main research interests are pattern recognition, image processing, machine vision and machine learning.Dr. Zhu has published 150+ peer-reviewed papers, including IEEE T-CSVT, IEEE T-NNLS, PR, AAAI, IJCAI, etc. He was awarded China National Excellence Doctoral Dissertation.