

# Consensus Graph Learning for Multi-view Clustering

Zhenglai Li, Chang Tang, Xinwang Liu, Xiao Zheng, Wei Zhang, En Zhu

**Abstract**—Multi-view clustering, which exploits the multi-view information to partition data into their clusters, has attracted intense attention. However, most existing methods directly learn a similarity graph from original multi-view features, which inevitably contain noises and redundancy information. The learned similarity graph is inaccurate and is insufficient to depict the underlying cluster structure of multi-view data. To address this issue, we propose a novel multi-view clustering method that is able to construct an essential similarity graph in a spectral embedding space instead of the original feature space. Concretely, we first obtain multiple spectral embedding matrices from the view-specific similarity graphs, and reorganize the gram matrices constructed by the inner product of the normalized spectral embedding matrices into a tensor. Then, we impose a weighted tensor nuclear norm constraint on the tensor to capture high-order consistent information among multiple views. Furthermore, we unify the spectral embedding and low rank tensor learning into a unified optimization framework to determine the spectral embedding matrices and tensor representation jointly. Finally, we obtain the consensus similarity graph from the gram matrices via an adaptive neighbor manner. An efficient optimization algorithm is designed to solve the resultant optimization problem. Extensive experiments on six benchmark datasets are conducted to verify the efficacy of the proposed method.

**Index Terms**—Multi-view clustering, consensus graph learning, weighted tensor nuclear norm.

## I. INTRODUCTION

With the development of information acquisition technologies, multimedia data, e.g. text, audio, image, video, are often captured from various sources or described by multiple features in numerous real-world applications [1], [2]. For example, the color, textures, and edges can be utilized to depict images in multimedia retrieval [1]. Multi-view video can be captured from diverse view points by multiple different cameras in the same scene and used in multi-view video summarization [2]. This kind of data is called multi-view data and consequently results in a battery of multi-view learning

methods, including cross-view domain learning [3], [4], multi-view clustering [5]–[9] and multi-view outlier detection [10], to list a few. Obtaining the semantic information of data is an important research topic in multimedia data mining. Multi-view clustering aims to capture the intrinsic categorical information of data, which is the key to analyze multimedia data, by exploring the multi-view feature of data in an unsupervised manner.

Spectral clustering, owing to its well-defined mathematical framework and partition ability on arbitrary shaped clusters, becomes one of the popular clustering algorithms. As a consequence, multi-view spectral clustering methods have been designed and applied to analyze multimedia data in recent years [11]–[15]. The multi-view spectral clustering methods usually contain two main steps: 1) constructing a shared similarity graph from multi-view data; 2) conducting the spectral clustering [16] on the similarity graph to obtain the final partition results. Due to heterogeneous sources of multimedia acquisition, the features of multi-view data are of redundancy, correlation, and diversity [17]. 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. To this end, Gao et al. [18] integrated the subspace representation learning and spectral clustering into a unified model to capture the consensus clustering structure. The work in [19] captures complementary information among multiple subspace representations by pursuing these representations to be diverse with a Hilbert-Schmidt independence criterion (HSIC) [20] regularization. In [21], an exclusive regularization is introduced to force the multiple subspace representations as diverse as possible for efficiently capturing the complementary information among multiple views. In [22], clustering and local structure learning are jointly performed to obtain an optimal graph. These methods pairwise investigate the view correlations to capture consistent and diverse information among multiple views. Unlikely, the tensor-based multi-view clustering methods which explore the high-order view correlations by reorganizing multiple representations into tensor form, have achieved promising performance and attracted much attention recently [14], [23]–[25].

Although these methods have improved the clustering performance from various aspects and achieved promising results, we observe that they learn the similarity graph in the original feature space in which the noises and redundancy information exist. Thus, the learned similarity graph is insufficient for clustering tasks. To address this issue, we propose a novel multi-view clustering method termed CGL that can construct

Manuscript received August 04, 2020; revised X XX, XXXX. The work was supported by the National Natural Science Foundation of China (NO. 61701451 and 62076228) ; Natural Natural Science Foundation of Hubei Province (NO. 2020CFB644) ; Key Laboratory of Information Perception and Systems for Public Security of MIIT (Nanjing University of Science and Technology) (No. 202007). (Corresponding author: Chang Tang.)

Z. Li and C. Tang are with the school of computer, China University of Geosciences, Wuhan, China. (E-mail: {yuezhenguan, tangchang}@cug.edu.cn).

X. Liu, X. Zheng, and E. Zhu are with the school of computer, National University of Defense Technology, Changsha, Hunan, China. (E-mail: {xinwangliu, enzhu}@nudt.edu.cn; xznudt@gmail.com).

W. Zhang is with Shandong Provincial Key Laboratory of Computer Networks, Shandong Computer Science Center (National Supercomputer Center in Jinan), Qilu University of Technology (Shandong Academy of Sciences), Jinan 250000, P.R. China. (E-mail: wzhang@qlu.edu.cn).

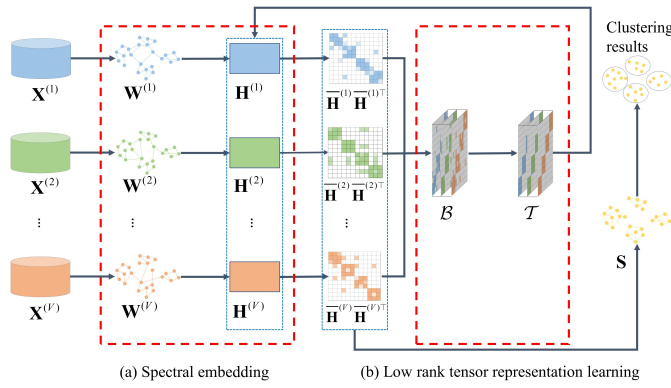


Fig. 1: Framework of the proposed CGL method. Multi-view similarity graphs  $\{\mathbf{W}^{(v)}\}_{v=1}^V$  are generated from multi-view data  $\{\mathbf{X}^{(v)}\}_{v=1}^V$  in advance. Multi-view embedded representations  $\{\mathbf{H}^{(v)}\}_{v=1}^V$  are obtained via (a) spectral embedding. To effectively capture the global consistency among multiple views, a low rank tensor  $\mathcal{T}$  is learned from a corrupted tensor  $\mathcal{B}$ , which is constructed by stacking the inner product of normalized embedded representations  $\{\bar{\mathbf{H}}^{(v)}\bar{\mathbf{H}}^{(v)\top}\}_{v=1}^V$  into a third-order tensor form. We further integrate the (a) spectral embedding and (b) low rank tensor representation learning into a unified optimization framework to achieve mutual promotion. Finally, the consensus graph  $\mathbf{S}$  can be learned in the embedded space.

an essential similarity graph in the spectral embedding space. Fig. 1 gives the flowchart of the proposed method. We stack multi-view gram matrices constructed by the inner product of normalized spectral embedding matrices into a third-order tensor. Ideally, the sample correlations are consistent across multiple gram matrices so that the third-order tensor is low rank. To this end, we impose a tensor low rank constraint to capture the high-order consistent information among multiple views. To improve the flexibility of the tensor nuclear norm, we introduce a weighted tensor nuclear norm by assigning different singular values with different weights. Furthermore, we integrate the spectral embedding and low rank tensor learning into a unified optimization framework to achieve mutual promotion. Finally, a consistent similarity graph can be constructed from multi-view normalized spectral embedding matrices via an adaptive neighbor graph learning manner. The contributions of this paper are summarized as follows,

- 1) We propose a novel multi-view clustering method that performs spectral embedding and tensor representation learning in a unified optimization framework. Compared with original features, the obtained spectral embedding representation is more beneficial to construct an intrinsic similarity graph for the clustering task.
- 2) We introduce a weighted tensor nuclear norm to capture the high-order consistent information among multiple views. By adaptively assigning different weights on singular values of the tensor, the weighting strategy improves the flexibility of the tensor nuclear norm in tensor low rank approximation.
- 3) We design an alternating optimization algorithm to solve

the proposed model. Furthermore, Extensive experiments on various benchmark datasets are conducted to verify 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 some notations and preliminaries, including the tensor nuclear norm and adaptive neighbor graph learning. The details of the proposed method and designed optimization algorithm are introduced in Section 4. Section 5 provides a series of experimental results, discussion, and model analysis. In Section 6, we provide a conclusion of this paper.

## II. RELATED WORK

Based on the way to construct the similarity graph, the multi-view clustering methods can be generally grouped into two categories, i.e., subspace segmentation based methods [8], [14], [18], [19], [21], [24]–[38] and graph based methods [22], [39]–[48].

Subspace segmentation aims to find multiple low dimensional subspaces embedded in a high dimensional space and partition the data points into their respective subspaces. Low rank subspace clustering (LRR) [49] and sparse subspace clustering (SSC) [50] are two representative single view clustering methods. Based on LRR and SSC, numerous multi-view clustering methods have been proposed. For example, In [18], the view-specific subspace representations and consensus cluster structures are learned simultaneously. Brbić et al. [28] simultaneously explored the low-rankness and sparsity of multi-view subspace representations, then learned a shared representation for clustering. To deal with large scale data, Kang et al. [30] proposed a linear order complexity multi-view clustering method which first learns multi-view anchor graphs and then integrates these graphs to obtain final clustering results. Zhang et al. [26] jointly learned the latent representation and subspace representations and combined the latent representation learning with a neural network to improve the generalization. Tang et al. [27] learned a joint affinity graph from multiple views for clustering with a rank constraint and diversity regularization. In [8], [14], [24], [25], [31], [33], multi-view subspace representations are reorganized into a third-order tensor and using the low rank tensor constraint to capture the high-order view correlations.

For graph based methods, they learn the similarity graph based on the Euclidean distance. In [39], a framework based on standard spectral learning is proposed to capture the consistent manifold among multiple graphs. In [40], a Laplacian rank constrained graph is learned as the centroid of multiple graphs with proper view confidences. In [41], a robust Markov chain method, in which a shared low-rank transition probability matrix is recovered from various transition probability matrices, is proposed for multi-view spectral clustering. In [42], the robust Markov chain method is extended into a tensor domain to capture high order view correlations. By minimizing the disagreement among multiple views, a consensus graph with the Laplacian rank constraint is learned in [43]. The work in [44] combines multiple graph structures to explore the underlying data geometric property. To capture the shared

graph structure of multiple views, Kun et al. [44] utilized the Hadamard product to integrate the numerous graphs into a global one for obtaining the joint edges among multiple graphs, then, recovered the graph structure from the global graph.

### III. NOTATIONS AND PRELIMINARIES

Throughout this paper, the capital case letters, boldface lower case letters, boldface capital letters and bold calligraphy letters are used to denote the entries, vectors, matrices, tensors, respectively, e.g.,  $X_{ij}$ ,  $\mathbf{x}$ ,  $\mathbf{X}$  and  $\mathcal{X}$ . For a third-order tensor  $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ ,  $\mathcal{X}(i, :, :)$ ,  $\mathcal{X}(:, j, :)$  and  $\mathcal{X}(:, :, k)$  denote the  $i$ -th horizontal,  $j$ -th lateral and  $k$ -th frontal slice of  $\mathcal{X}$ , respectively.  $\text{conj}(\mathcal{X})$  and  $\mathcal{X}^*$  represents the complex conjugate and conjugate transpose of  $\mathcal{X}$ . We use  $\lceil t \rceil$  to denote as the nearest integer greater than or equal to  $t$ .  $\text{Tr}(\mathbf{X}) = \sum_i X_{ii}$  denotes the trace of a square matrix  $\mathbf{X}$ .  $\mathbf{I}_k$  is used to represent an  $k \times k$  identity matrix. We use  $\|\mathbf{X}\|_F = \sqrt{\sum_{ij} X_{ij}^2}$  and  $\|\mathcal{X}\|_F = \sqrt{\sum_{ijk} \mathcal{X}(i, j, k)^2}$  to denote the Frobenius norm of the matrix  $\mathbf{X}$  and tensor  $\mathcal{X}$ , respectively.

#### A. Tensor nuclear norm based on t-SVD

The tensor singular value decomposition (t-SVD) of a tensor  $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$  is represented as [51],

$$\mathcal{X} = \mathcal{U} * \mathcal{S} * \mathcal{V} \quad (1)$$

where  $\mathcal{U} \in \mathbb{R}^{n_1 \times n_1 \times n_3}$ ,  $\mathcal{S} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$  are two orthogonal tensors,  $\mathcal{V} \in \mathbb{R}^{n_2 \times n_2 \times n_3}$  is an f-diagonal tensor, and  $*$  represents the tensor-to-tensor product.

Then, the tensor nuclear norm (TNN) of the tensor  $\mathcal{X}$  is given as [51],

$$\|\mathcal{X}\|_* = \sum_{i=1}^r \mathcal{S}(i, i, 1) = \sum_{i=1}^r \sum_{j=1}^{n_3} \bar{\mathcal{S}}(i, i, j) \quad (2)$$

where  $r$  is the tensor tubal rank of  $\mathcal{X}$ ,  $\bar{\mathcal{S}} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$  is denoted as the result of Discrete Fourier Transformation (DFT) on  $\mathcal{X}$  along the 3-rd dimension, i.e.,  $\bar{\mathcal{S}} = \text{fft}(\mathcal{X}, [], 3)$ . [51] gives the detailed definitions of those tensors and operators.

#### B. Adaptive neighbor graph learning

Given a data matrix  $\mathbf{X} \in \mathbb{R}^{n \times d}$ ,  $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ . The work in [52] proposes an adaptive neighbor graph learning method by considering the probabilistic neighbors. The connect probability of two data points can be treated as the similarity between them. The objective function of this method can be formulated as follows:

$$\begin{aligned} \min_{\mathbf{s}_i} \quad & \sum_{j=1}^n (\|\mathbf{x}_i - \mathbf{x}_j\|_2^2 \mathbf{s}_{ij} + \gamma \|S_{ij}\|_2^2) \\ \text{s.t.} \quad & \mathbf{s}_{ij} \geq 0, \mathbf{s}_i \mathbf{1}_n = 1 \end{aligned} \quad (3)$$

where  $\mathbf{x}_i$  and  $\mathbf{x}_j$  are the  $i$ -th and  $j$ -th data points,  $S_{ij}$  denotes the connect probability between data points  $\mathbf{x}_i$  and  $\mathbf{x}_j$ ,  $\gamma$  is a

balance parameter. The second term is used to avoid the trivial solution. Eq. (3) can be rewritten as,

$$\begin{aligned} \min_{\mathbf{s}_i} \quad & \|\mathbf{s}_i - \frac{\mathbf{d}_i^x}{2\gamma}\|_2^2 \\ \text{s.t.} \quad & \mathbf{s}_{ij} \geq 0, \mathbf{s}_i \mathbf{1}_n = 1 \end{aligned} \quad (4)$$

where  $\mathbf{d}_i^x$  is the  $i$ -th row of the distance matrix  $\mathbf{D}^x$  with  $D_{ij}^x = \|\mathbf{x}_i - \mathbf{x}_j\|_2^2$ . Then, the  $k$ -sparse similarity graph  $\mathbf{S}$  is obtained as,

$$S_{ij} = \frac{\bar{D}_{i,k+1}^x - \bar{D}_{ij}^x}{k \bar{D}_{i,k+1}^x - \sum_{j=1}^k \bar{D}_{ij}^x} \quad (5)$$

where  $\bar{\mathbf{d}}_i^x$  is given by ordering  $\mathbf{d}_i^x$  from small to large.

### IV. PROPOSED METHOD

In this section, we present the proposed multi-view clustering model and then solve the resultant optimization problem by designing an alternating iterative algorithm.

#### A. Problem formulation

Given a set of data matrices  $\{\mathbf{X}^{(v)}\}_{v=1}^V$ ,  $\mathbf{X}^{(v)} \in \mathbb{R}^{n \times d_v}$ , where  $d_v$  denotes the dimension of the features in  $v$ -th view,  $n$  is the number of data samples,  $V$  is the number of views. Based on the adaptive neighbor graph learning manner in Eq. (3), the work in [22] proposes a multi-view adaptive neighbor graph learning method which learns a Laplacian rank constrained consensus graph from multi-view data by allocating idea view weights. The model can be formulated as follows,

$$\begin{aligned} \min_{\mathbf{S}} \quad & \sum_{v=1}^V w_v \sum_{i=1}^n \sum_{j=1}^n \|\mathbf{x}_i^{(v)} - \mathbf{x}_j^{(v)}\|_2^2 S_{ij} + \gamma \|\mathbf{S}\|_F^2 \\ \text{s.t.} \quad & \mathbf{s}_{ij} \geq 0, \mathbf{s}_i \mathbf{1}_n = 1, \text{rank}(\mathbf{L}_s) = n - c, \\ & w_v = \frac{1}{2 \sqrt{\sum_{i=1}^n \sum_{j=1}^n \|\mathbf{x}_i^{(v)} - \mathbf{x}_j^{(v)}\|_2^2 S_{ij}}} \end{aligned} \quad (6)$$

where  $w_v$  is the view weight of  $v$ -th view,  $\mathbf{S}$  is the consensus similarity graph,  $\mathbf{L}_s = \mathbf{D} - \mathbf{S}$  represents the Laplacian matrix of  $\mathbf{S}$ ,  $\mathbf{D}$  is a diagonal matrix with  $D_{ii} = \sum_{j=1}^n S_{ij}$ .

In practice, noises and redundancy information are usually mixed in the original features. Thus, the learned consensus similarity graph may be inaccurate in Eq. (6). To address this issue, we learn the adaptive neighbor graph in a new low dimensional embedding space instead of original feature space. Based on Eq. (3), the adaptive neighbor graph can be obtained by solving,

$$\begin{aligned} \min_{\mathbf{S}} \quad & \sum_{v=1}^V \sum_{i=1}^n \sum_{j=1}^n \|\bar{\mathbf{h}}_i^{(v)} - \bar{\mathbf{h}}_j^{(v)}\|_2^2 S_{ij} + \gamma \|\mathbf{S}\|_F^2 \\ \text{s.t.} \quad & \mathbf{s}_{ij} \geq 0, \mathbf{s}_i \mathbf{1}_n = 1. \end{aligned} \quad (7)$$

where  $\bar{\mathbf{h}}_i^{(v)}$  and  $\bar{\mathbf{h}}_j^{(v)}$  are the  $i$ -th and  $j$ -th rows of normalized spectral embedding matrix  $\bar{\mathbf{H}}^{(v)}$ .  $\bar{\mathbf{H}}^{(v)}$  is obtained by normalizing the rows of  $\mathbf{H}^{(v)}$  to achieve unit Euclidean length, i.e.,  $\bar{\mathbf{h}}_i^{(v)} = \frac{\mathbf{h}_i^{(v)}}{\sqrt{\mathbf{h}_i^{(v)\top} \mathbf{h}_i^{(v)}}}$ . After the normalization step, the

normalized spectral embedding matrix contains  $k$ -dimensional representations of the samples on a unit sphere, and Euclidean distance based similarity can well depict the cluster structure among samples in such a case.  $\mathbf{H}^{(v)}$  is the  $v$ -th view spectral embedding matrix and obtained by performing spectral clustering on view-specific similarity graph  $\mathbf{W}^{(v)}$  as,

$$\max_{\mathbf{H}^{(v)}} \text{Tr}(\mathbf{H}^{(v)\top} \mathbf{A}^{(v)} \mathbf{H}^{(v)}) \quad s.t. \quad \mathbf{H}^{(v)} \mathbf{H}^{(v)\top} = \mathbf{I}_c. \quad (8)$$

where  $\mathbf{A}^{(v)}$  is computed as  $\mathbf{D}^{(v)-0.5} \mathbf{W}^{(v)} \mathbf{D}^{(v)-0.5}$ ,  $\mathbf{D}^{(v)}$  is a diagonal matrix with  $D_{ii}^{(v)} = \sum_{j=1}^n W_{ij}^{(v)}$ .

According to Eq. (5), the obtained similarity graph  $\mathbf{S}$  of Eq. (7) is largely depend on the distance matrices  $\mathbf{D}^{h(v)}$ , i.e.,  $D_{ij}^{h(v)} = \|\bar{\mathbf{h}}_i^{(v)} - \bar{\mathbf{h}}_j^{(v)}\|_2^2 = 2 - \bar{\mathbf{h}}_i^{(v)} \bar{\mathbf{h}}_j^{(v)}$ . Thus, the similarity graph learning problem is transferred into learning robust and comprehensive distance matrices  $\mathbf{D}^{h(v)}$ . In Eq. (8), the view correlations are not taken into consideration, resulting that multiple distance matrices lack of global consistency. By reorganizing  $\{\bar{\mathbf{H}}^{(v)} \bar{\mathbf{H}}^{(v)\top}\}_{v=1}^V$  into a third-order tensor  $\mathcal{B} \in \mathbb{R}^{n \times V \times n}$ , the sample correlations are consistent across multiple views so that the tensor  $\mathcal{B}$  should be low rank. To this end, we formulate the optimization goal as,

$$\min_{\mathbf{H}^{(v)}, \mathcal{T}} -\lambda \sum_{v=1}^V \text{Tr}(\mathbf{H}^{(v)\top} \mathbf{A}^{(v)} \mathbf{H}^{(v)}) + \frac{1}{2} \|\mathcal{B} - \mathcal{T}\|_F^2 + \tau \|\mathcal{T}\|_* \quad s.t. \quad \mathbf{H}^{(v)\top} \mathbf{H}^{(v)} = \mathbf{I}_c. \quad (9)$$

where  $\lambda$  is a balance parameter,  $\tau$  is singular values threshold,  $\mathcal{T} \in \mathbb{R}^{n \times V \times n}$  is a tensor. In Eq. (9), the first term is the spectral embedding term. The second and third terms are used to capture the primary component of tensor  $\mathcal{B}$  and regularize multiple gram matrices  $\{\bar{\mathbf{H}}^{(v)} \bar{\mathbf{H}}^{(v)\top}\}_{v=1}^V$  to be consistent with each other. The low rank tensor  $\mathcal{T}$  can be obtained by using the tensor singular value thresholding (t-SVT) operator as follows [51],

$$\mathcal{D}_\tau(\mathcal{B}) = \mathcal{U} * \mathcal{S}_\tau * \mathcal{V}^* \quad (10)$$

where  $\mathcal{S}_\tau = \text{ifft}((\bar{\mathcal{S}} - \tau)_+, [\cdot], 3)$ ,  $t_+ = \max(t, 0)$ .

In Eq. (10), the singular values are equally shrunk with the same singular values threshold  $\tau$ . However, the larger singular values quantify the information of the underlying principal directions and should be less shrunk. The over-penalization of large singular values would obtain a biased solution and be not conducive to mining the primary components of the tensor. Therefore, we introduce a weighted tensor nuclear norm to improve the flexibility of the tensor nuclear norm as follows,

$$\|\mathcal{T}\|_{w,*} = \sum_{i=1}^r \sum_{j=1}^n w_i^{(j)} \bar{\mathcal{S}}(i, j) \quad (11)$$

where  $w_i^{(j)}$  is a singular value weight.

Finally, we formulate the final objective function as follows,

$$\min_{\mathbf{H}^{(v)}, \mathcal{T}} -\lambda \sum_{v=1}^V \text{Tr}(\mathbf{H}^{(v)\top} \mathbf{A}^{(v)} \mathbf{H}^{(v)}) + \frac{1}{2} \|\mathcal{B} - \mathcal{T}\|_F^2 + \|\mathcal{T}\|_{w,*} \quad s.t. \quad \mathbf{H}^{(v)\top} \mathbf{H}^{(v)} = \mathbf{I}_c. \quad (12)$$

By solving the problem in Eq. (12), the consensus graph  $\mathbf{S}$  can be learned from the gram matrices  $\{\bar{\mathbf{H}}^{(v)} \bar{\mathbf{H}}^{(v)\top}\}_{v=1}^V$  via Eq. (7). In the proposed CGL method, the distribution of noises and redundant information are treated as diverse among multiple views. By capturing the global consistency among multiple views, the noise and redundant information can be effectively filtered out. Thus, the learned embedded representation is more beneficial to construct an intrinsic similarity graph for the clustering task.

### B. Optimization algorithm

To solving the problem in Eq. (12), we design an optimization algorithm in an alternate iteration manner. Thus, the problem can be transformed into two subproblems, i.e.,  $\mathbf{H}^{(v)}$ -subproblem and  $\mathcal{T}$ -subproblem.

**$\mathbf{H}^{(v)}$ -subproblem:** We first fix variable  $\mathcal{T}$ , then unfold two tensors  $\mathcal{B}$  and  $\mathcal{T}$  into matrix form. By dropping other unrelated terms, Eq. (12) can be rewritten as,

$$\min_{\mathbf{H}^{(v)}} -\lambda \text{Tr}(\mathbf{H}^{(v)\top} \mathbf{A}^{(v)} \mathbf{H}^{(v)}) + \frac{1}{2} \|\bar{\mathbf{H}}^{(v)} \bar{\mathbf{H}}^{(v)\top} - \mathbf{T}^{(v)}\|_F^2 \quad s.t. \quad \mathbf{H}^{(v)\top} \mathbf{H}^{(v)} = \mathbf{I}_c. \quad (13)$$

where  $\mathbf{T}^{(v)}$  is the  $v$ -th lateral slice of tensor  $\mathcal{T}$ , i.e.,  $\mathbf{T}^{(v)} = \mathcal{T}(:, v, :)$ . Eq. (13) can be further rewritten as,

$$\min_{\mathbf{H}^{(v)}} -\lambda \text{Tr}(\mathbf{H}^{(v)\top} \mathbf{A}^{(v)} \mathbf{H}^{(v)}) + \frac{1}{2} \text{Tr}(\bar{\mathbf{H}}^{(v)} \bar{\mathbf{H}}^{(v)\top} \bar{\mathbf{H}}^{(v)} \bar{\mathbf{H}}^{(v)\top}) - \frac{1}{2} \text{Tr}(\bar{\mathbf{H}}^{(v)} \bar{\mathbf{H}}^{(v)\top} (\mathbf{T}^{(v)} + \mathbf{T}^{(v)\top})) \quad s.t. \quad \mathbf{H}^{(v)\top} \mathbf{H}^{(v)} = \mathbf{I}_c. \quad (14)$$

Let  $\mathbf{P}^{(v)} \in \mathbb{R}^{n \times n}$  be a diagonal matrix. Its elements are obtained as

$$P_{ij}^{(v)} = \begin{cases} \frac{1}{\sqrt{\mathbf{h}_i^{(v)\top} \mathbf{h}_i^{(v)}}}, & \text{if } i = j \\ 0, & \text{otherwise} \end{cases} \quad (15)$$

Then, we have the following equation,

$$\bar{\mathbf{H}}^{(v)} = \mathbf{P}^{(v)} \mathbf{H}^{(v)} \quad (16)$$

By integrating Eq. (16) into Eq. (14), the optimization problem can be further rewritten as,

$$\max_{\mathbf{H}^{(v)}} \text{Tr}(\mathbf{H}^{(v)\top} \mathbf{G}^{(v)} \mathbf{H}^{(v)}) \quad s.t. \quad \mathbf{H}^{(v)\top} \mathbf{H}^{(v)} = \mathbf{I}_c. \quad (17)$$

where  $\mathbf{G}^{(v)} = \lambda \mathbf{A}^{(v)} + \frac{1}{2} \mathbf{P}^{(v)} (\mathbf{T}^{(v)} + \mathbf{T}^{(v)\top}) \mathbf{P}^{(v)} - \frac{1}{2} \mathbf{P}^{(v)} \bar{\mathbf{H}}^{(v)} \bar{\mathbf{H}}^{(v)\top} \mathbf{P}^{(v)}$ . The optimal solution of  $\mathbf{H}^{(v)}$  can be obtain by taking the eigenvectors corresponding to the  $c$  largest eigenvalues of  $\mathbf{G}^{(v)}$ .

**$\mathcal{T}$ -subproblem:** when we fix  $\{\mathbf{H}^{(v)}\}_{v=1}^V$  and drop other unrelated term, Eq. (12) can be rewritten as,

$$\min_{\mathcal{T}} \frac{1}{2} \|\mathcal{B} - \mathcal{T}\|_F^2 + \|\mathcal{T}\|_{w,*} \quad (18)$$



For a tensor  $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ , we have  $\|\mathcal{X}\|_F = \frac{1}{\sqrt{n_3}} \|\bar{\mathcal{X}}\|_F$ . Then, Eq. (18) is equivalent to

$$\min_{\bar{\mathcal{T}}(:, :, j)} \frac{1}{n} \sum_{j=1}^n \left( \frac{1}{2} \|\bar{\mathcal{B}}(:, :, j) - \bar{\mathcal{T}}(:, :, j)\|_F^2 + \|\bar{\mathcal{T}}(:, :, j)\|_{w,*} \right) \quad (19)$$

where  $\bar{\mathcal{T}}(:, :, j)$  and  $\bar{\mathcal{B}}(:, :, j)$  are the  $j$ -th slices of  $\bar{\mathcal{T}}$  and  $\bar{\mathcal{B}}$ , respectively.  $\bar{\mathcal{T}}$  and  $\bar{\mathcal{B}}$  denote the result of DFT on  $\mathcal{T}$  and  $\mathcal{B}$  along the 3-rd dimension, i.e.,  $\bar{\mathcal{T}} = \text{fft}(\mathcal{T}, [], 3)$  and  $\bar{\mathcal{B}} = \text{fft}(\mathcal{B}, [], 3)$ .

By using the reweighting strategy in [53],  $\bar{\mathcal{T}}(:, :, j)$  has a close-form solution,

$$\bar{\mathcal{T}}(:, :, j) = \bar{\mathcal{U}}(:, :, j) * \bar{\mathcal{S}}(:, :, j) * \bar{\mathcal{V}}(:, :, j)^* \quad (20)$$

where  $\bar{\mathcal{B}}(:, :, j) = \bar{\mathcal{U}}(:, :, j) * \bar{\mathcal{S}}(:, :, j) * \bar{\mathcal{V}}(:, :, j)^*$  is the SVD of  $\bar{\mathcal{B}}(:, :, j)$ ,  $\bar{\mathcal{S}}(:, :, j)$  is obtained as,

$$\bar{\mathcal{S}}(i, i, j) = \begin{cases} 0, & \text{if } c_2 < 0 \\ \frac{c_1 + \sqrt{c_2}}{2}, & \text{if } c_2 \geq 0 \end{cases} \quad (21)$$

where  $c_1 = \bar{\mathcal{S}}(i, i, j) - \epsilon$ ,  $c_2 = (\bar{\mathcal{S}}(i, i, j) + \epsilon)^2 - 4C$ .  $\epsilon$  is a small enough positive value to make the inequality  $\epsilon < \min(\sqrt{C}, \frac{C}{\bar{\mathcal{S}}(i, i, j)})$  hold,  $C$  is a regularization parameter to set the weight  $w_i^{(j)}$ , i.e.,  $w_i^{(j)} = \frac{C}{\bar{\mathcal{S}}(i, i, j) + \epsilon}$ .

1 and 2, respectively. The computation complexity and empirical convergence analysis are given in the next section.

---

#### Algorithm 1: Solution to Eq. (18)

---

**Input:** Tensor  $\mathcal{B} \in \mathbb{R}^{n \times V \times n}$  and regularization parameter  $C$ ;  
**Output:** Tensor  $\mathcal{T} \in \mathbb{R}^{n \times V \times n}$ ;  
1 Compute  $\bar{\mathcal{B}} = \text{fft}(\mathcal{B}, [], 3)$ ;  
2 Perform reweighted strategy on each frontal slice of  $\bar{\mathcal{B}}$  by;  
3 **for**  $j = 1, \dots, \lceil \frac{n+1}{2} \rceil$  **do**  
4    $\bar{\mathcal{U}}(:, :, j) * \bar{\mathcal{S}}(:, :, j) * \bar{\mathcal{V}}(:, :, j)^* = \text{SVD}(\bar{\mathcal{B}}(:, :, j))$ ;  
5   Obtain  $\bar{\mathcal{S}}(:, :, j)$  via Eq. (21);  
6    $\bar{\mathcal{T}}(:, :, j) = \bar{\mathcal{U}}(:, :, j) * \bar{\mathcal{S}}(:, :, j) * \bar{\mathcal{V}}(:, :, j)^*$ ;  
7 **end**  
8 **for**  $j = \lceil \frac{n+1}{2} \rceil + 1, \dots, n$  **do**  
9    $\bar{\mathcal{T}}(:, :, j) = \text{conj}(\bar{\mathcal{T}}(:, :, n - j + 2))$ ;  
10 **end**  
11 Compute  $\bar{\mathcal{T}} = \text{ifft}(\bar{\mathcal{T}}, [], 3)$ .

---

We summarize the weighted tensor nuclear norm minimization of the  $\mathcal{T}$ -subproblem and whole optimization procedure of the proposed method in Algorithm

### C. Discussion

In this section, some discussions are given to further understand the proposed method.

$$\min_{\mathcal{T}} \frac{1}{2} \|\mathcal{B} - \mathcal{T}\|_F^2 + \|\mathcal{T}\|_{w,*} \quad (22)$$

---

#### Algorithm 2: Consensus graph learning for multi-view clustering

---

**Input:** Multi-view data  $\{\mathbf{X}^{(v)}\}_{v=1}^V$ , nearest neighbor  $k$ , balance parameter  $\lambda$ , regularization parameter  $C$ , clustering number  $c$  and  $\epsilon = 0.0001$ ;  
**Output:** Consensus graph  $\mathbf{S}$  and clustering results  $\mathcal{Y}$ ;  
1 Initialize  $t = 1$ ,  $\{\mathbf{H}^{(v)} = \mathbf{0}\}_{v=1}^V$ ,  $\{\bar{\mathbf{H}}^{(v)} = \mathbf{0}\}_{v=1}^V$ ,  $\{\mathbf{P}^{(v)} = \mathbf{0}\}_{v=1}^V$ ,  $\{\mathcal{T}(:, :, j) = \mathbf{0}\}_{j=1}^n$ ;  
2 Compute view-specific graphs  $\{\mathbf{W}^{(v)}\}_{v=1}^V$  via Eq. (3);  
3 Compute  $\{\mathbf{A}^{(v)}\}_{v=1}^V$ ;  
4 **while not converged do**  
5   Update  $\{\mathbf{H}^{(v)}\}_{v=1}^V$  via Eq. (17);  
6   Update  $\{\mathbf{P}^{(v)}\}_{v=1}^V$  and  $\{\bar{\mathbf{H}}^{(v)}\}_{v=1}^V$  via Eq. 15 and Eq. 16, respectively;  
7   Obtain  $\bar{\mathcal{B}} = \Phi(\bar{\mathbf{H}}^{(1)} \bar{\mathbf{H}}^{(1)\top}, \dots, \bar{\mathbf{H}}^{(V)} \bar{\mathbf{H}}^{(V)\top})$ ;  
8   Update  $\mathcal{T}$  via Algorithm 1;  
9   Compute objective function value  $\text{obj}(t)$ ;  
10   Check convergence condition  $\frac{|\text{obj}(t) - \text{obj}(t-1)|}{\text{obj}(t-1)} < \epsilon$ ;  
11    $t = t + 1$ ;  
12 **end**  
13 Compute consensus graph  $\mathbf{S}$  via Eq. (7);  
14 Employ the spectral clustering on the obtained consensus similarity graph  $\mathbf{S}$ ;  
15 return the consensus graph  $\mathbf{S}$ , clustering results  $\mathcal{Y}$ ;

---

In Eq. (22), the  $v$ th slice of tensor  $\mathcal{B}$ , i.e.  $\mathcal{B}(:, v, :)$ , represents the sample dissimilarities of  $v$ th view of multi-view data. Due to the noises and redundancy information mixed in original features, the obtained sample dissimilarities in each slice of tensor  $\mathcal{B}$  may be inaccurate. And tensor  $\mathcal{B}$  can be regarded as a degraded tensor with noise information. Eq. (22) is the tensor robust principal component analysis [51] with a weighted tensor nuclear norm. It aims to capture the principal components of a tensor and can be utilized to filter the noise sample dissimilarities and reduce view redundancy. Subsequently, the sample correlations of multi-view data can be depicted more accurate by the learned each slice of tensor  $\mathcal{T}$  than that of tensor  $\mathcal{B}$ .

$$\max_{\mathbf{H}^{(v)}} \text{Tr}(\mathbf{H}^{(v)\top} \mathbf{G}^{(v)} \mathbf{H}^{(v)}) \text{ s.t. } \mathbf{H}^{(v)\top} \mathbf{H}^{(v)} = \mathbf{I}_c. \quad (23)$$

where  $\mathbf{G}^{(v)} = \lambda \mathbf{A}^{(v)} + \frac{1}{2} \mathbf{P}^{(v)} (\mathbf{T}^{(v)} + \mathbf{T}^{(v)\top}) \mathbf{P}^{(v)} - \frac{1}{2} \mathbf{P}^{(v)} \bar{\mathbf{H}}^{(v)} \bar{\mathbf{H}}^{(v)\top} \mathbf{P}^{(v)}$ .  $\mathbf{A}^{(v)}$  is the  $v$ th view normalized similarity graph which is obtained from the  $v$ th original features. Consequently, it may contain some inaccurate sample correlations. By combining  $\mathbf{T}^{(v)}$  with  $\mathbf{A}^{(v)}$ , the inaccurate sample correlations can be suppressed in the obtained matrix  $\mathbf{G}^{(v)}$ . Then, the process of SVD in Eq. 23 transforms the representations of multi-view data  $\{\mathbf{X}^{(v)}\}$  from the original space to an embedded space where the cluster properties are enhanced in this space. Thus, the obtained embedded features are of better discriminability and are more beneficial to construct an intrinsic similarity graph for the clustering task. Furthermore, the clustering ability of the learned embedded features will be verified in the next section.

## V. EXPERIMENTS

In this section, we evaluate the clustering performance of the proposed method on six widely used multi-view benchmark datasets and conduct a series of model analysis.

### A. Experimental settings

1) *Datasets*: We adopt six benchmark datasets in the experiments. The details of the datasets are as follows,

**MSRCV1**: It consists of 210 scene recognition images belonging to seven classes. Each image is represented by six different feature sets, i.e., 256 dimension LBP, 100 dimension HOG, 512 dimension GIST, 48 dimension Color Moment, 1302 dimension CENTRIST, and 210 dimension SIFT.

**ORL**<sup>1</sup>: It contains 400 face images of 40 individuals under different lighting, times, and facial details. Three feature sets, including 4096 dimension intensity, 3304 dimension LBP, and 6750 dimension Gabor are utilized.

**20newsgroups**<sup>2</sup>: It is a newsgroups documents dataset consisting of five classes, 500 documents in total. Three different feature sets are generated by three pre-processed methods.

**100leaves**<sup>3</sup>: This set contains 1600 samples of 100 plant species. Shape descriptor, texture histogram and fine scale margin are extracted to depict each sample.

**COIL20** [54]: It consists of 1400 images of 20 classes. For each image, the 1024 dimension intensity, 3304 dimension LBP and 6750 dimension Gabor features are extracted.

**handwritten**: 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 dimension FOU, 216 dimension FAC, 64 dimension KAR, 240 dimension Pix, 47 dimension ZER, and six MOR.

2) *Compared methods*: To verify the superiority of the proposed method, two single view baselines and seven recently proposed multi-view clustering methods are used in the experiments. In detail, the compared methods are as follows,

**SC** [16]: Spectral clustering.

**LRR** [49]: This method constructs the subspace representation matrix by using nuclear norm constraint.

**MLAN** [22]: This method automatically captures the view weights and simultaneously learns a Laplacian rank constrained graph for clustering

**MCGC** [43]: This method employs a co-regularization term to reduce the disagreements among multiple views. Meanwhile, it learns a Laplacian rank constrained consensus graph from multiple spectral embedding matrices.

**GMC** [55]: This method integrates adaptive neighbor graph learning and multiple graphs fusing into a unified framework to learn a Laplacian rank constrained graph for clustering.

**SM2SC** [56]: This method extracts intrinsic components from view-specific subspace representations by using a variable splitting scheme and a multiplicative decomposition scheme.

**LT-MSC** [24]: This method learns low rank tensor representation by stacking multiple subspace representation into tensor form and making the three unfolding modes of the tensor be low rank.

**t-SVD-MS** [25]: This method stacks multiple subspace representations into tensor form and learn low rank tensor representation with a t-SVD based tensor nuclear norm.

**ETLMSC** [42]: This method integrates multiple transition probability matrices into a tensor and learns essential information from the tensor with a t-SVD based tensor and a  $l_{2,1}$  norm. Then, It conducts the Markov chain-based spectral clustering on the learned essential transition probability matrix to obtain the final clustering results.

3) *Parameter settings and evaluation metrics*: SC and LRR are two single view clustering baselines. We perform them on each single view and report the best clustering performance. For SC, the similarity graph is obtain via Eq. (3) with nearest neighbor  $k = 15$ . For LRR, we select the parameter from  $[10^{-3}, 10^{-2}, \dots, 10^2, 10^3]$ . For MLAN and GMC, the nearest neighbors are set as  $k = 9$  and  $k = 15$  as the settings in their paper. For MCGC, the nearest neighbor is set as  $k = 15$  and a regularization parameter is chosen from  $[0.6 : 5 : 100]$ . For SM2SC, we choose three regularization parameters from  $[0.1, 0.15, 0.2, 0.3, 0.4, 0.5, 1, 10, 40, 100]$ ,  $[0.1, 0.5, 1, 1.5, 2]$  and  $[0.05, 0.1, 0.4, 1, 5]$ , respectively. For LT-MSC, we select a regularization parameter from  $[0 : 0.05 : 0.5, 10 : 10 : 100]$ . For t-SVD-MS and ETLMSC, the regularization parameters are set within the range  $[0.1 : 0.1 : 2]$  and  $[10^{-4} : 10^{-4} : 10^{-3}, 10^{-3} : 10^{-3} : 10^{-2}, \dots, 10^1 : 10^1 : 10^2]$ , respectively. For our proposed CGL method, the nearest neighbor is set as  $k = 15$ . two parameters  $\lambda$  and  $C$  searched from  $[1, 5, 10, 50, 100, 500, 1000, 5000]$  with grid search strategy. For a fair comparison, we repeat each experiment 20 times and report mean values 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.

### B. Experimental results

1) *Clustering performance*: The clustering performance measured by seven evaluation metrics of all methods on six benchmark datasets is reported in TABLE I. From the results, we obtain the following observations:

(1) The clustering results of our proposed method significantly outperform that of the compared methods. Taking the MSRCV1 dataset as an example, it exceeds the second best SM2SC 5.24, 10.66, and 5.24 percent in terms of ACC, NMI, and Purity, respectively. This validates the advantages and effectiveness of our proposed method. The primary reason are that i) the proposed method learns a similarity graph based on the spectral embedding matrices, and ii) simultaneously conducting spectral embedding and tensor representation learning can obtain superior embedded features.

<sup>1</sup><http://www.uk.research.att.com/facedatabase.html>

<sup>2</sup><http://lig-membres.imag.fr/grimal/data.html>

<sup>3</sup><https://archive.ics.uci.edu/ml/datasets/One-hundred+plant+species+leaves+data+set>

TABLE I: Clustering performance on six 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
MSRCV1	SC	0.6684±0.0000	0.6354±0.0000	0.7051±0.0000	0.7109±0.0000	0.6118±0.0000	0.7548±0.0024	0.7810±0.0000
	LRR	0.5434±0.0099	0.5354±0.0093	0.8914±0.0882	0.5561±0.0055	0.4686±0.0114	0.6793±0.0035	0.6810±0.0027
	MLAN	0.6858±0.0000	0.6111±0.0000	0.7813±0.0000	0.7629±0.0000	0.6278±0.0000	0.7238±0.0000	0.7905±0.0000
	MCGC	0.6857±0.0000	0.6602±0.0000	0.7468±0.0000	0.7375±0.0000	0.6328±0.0000	0.7571±0.0000	0.8048±0.0000
	GMC	0.7997±0.0000	0.7856±0.0000	0.8144±0.0000	0.8200±0.0000	0.7668±0.0000	0.8952±0.0000	0.8952±0.0000
	SM2SC	0.8027±0.0003	0.7994±0.0002	0.8060±0.0004	0.8001±0.0025	0.7708±0.0003	0.8952±0.0000	0.8952±0.0000
	LT-MSC	0.7376±0.0006	0.7270±0.0011	0.7484±0.0000	0.7560±0.0002	0.6946±0.0007	0.8429±0.0000	0.8429±0.0000
	t-SVD-MS	0.7076±0.0013	0.6843±0.0019	0.7366±0.0002	0.7347±0.0015	0.6584±0.0015	0.8095±0.0000	0.8095±0.0000
	ETLMSC	0.6152±0.0071	0.5969±0.0068	0.6347±0.0075	0.6257±0.0061	0.5510±0.0083	0.7376±0.0046	0.7567±0.0046
	CGL	<b>0.8945±0.0000</b>	<b>0.8914±0.0000</b>	<b>0.8975±0.0000</b>	<b>0.8883±0.0000</b>	<b>0.8774±0.0000</b>	<b>0.9476±0.0000</b>	<b>0.9476±0.0000</b>
ORL	SC	0.7446±0.0182	0.6998±0.0248	0.7960±0.0149	0.9141±0.0059	0.7383±0.0187	0.7950±0.0240	0.8339±0.0156
	LRR	0.7657±0.0175	0.7169±0.0250	0.8220±0.0121	0.9255±0.0054	0.7599±0.0180	0.8151±0.0188	0.8476±0.0119
	MLAN	0.3544±0.0000	0.2347±0.0000	0.7233±0.0000	0.8312±0.0000	0.3316±0.0000	0.6850±0.0000	0.7350±0.0000
	MCGC	0.5644±0.0000	0.4780±0.0000	0.7606±0.0000	0.8656±0.0000	0.5525±0.0000	0.7200±0.0000	0.7800±0.0000
	GMC	0.3599±0.0000	0.2321±0.0000	0.8011±0.0000	0.8571±0.0000	0.3367±0.0000	0.6325±0.0000	0.7150±0.0000
	SM2SC	0.6419±0.0228	0.6091±0.0255	0.7167±0.0152	0.8539±0.0096	0.6332±0.0234	0.7624±0.0221	0.7813±0.0199
	LT-MSC	0.7663±0.0258	0.7203±0.0310	0.8188±0.0209	0.9207±0.0092	0.7605±0.0265	0.8162±0.0193	0.8481±0.0175
	t-SVD-MS	0.7679±0.0220	0.7303±0.0226	0.8216±0.0241	0.9221±0.0068	0.7623±0.0225	0.8209±0.0192	0.8460±0.0154
	ETLMSC	0.7024±0.0323	0.6639±0.0318	0.7459±0.0358	0.8903±0.0152	0.6951±0.0331	0.7734±0.0239	0.8021±0.0235
	CGL	<b>0.8584±0.0060</b>	<b>0.8446±0.0074</b>	<b>0.8727±0.0050</b>	<b>0.9454±0.0021</b>	<b>0.8551±0.0061</b>	<b>0.8996±0.0051</b>	<b>0.9074±0.0038</b>
20newsgroups	SC	0.6078±0.0000	0.5276±0.0000	0.7167±0.0000	0.5643±0.0000	0.4915±0.0000	0.6600±0.0000	0.7040±0.0000
	LRR	0.6295±0.0002	0.6163±0.0001	<b>0.9840±0.0000</b>	0.5464±0.0013	0.5353±0.0002	0.7840±0.0000	0.7840±0.0000
	MLAN	0.5237±0.0000	0.4222±0.0000	0.6895±0.0000	0.5248±0.0000	0.3682±0.0000	0.5900±0.0000	0.6500±0.0000
	MCGC	0.6170±0.0000	0.5079±0.0000	0.7859±0.0000	0.6495±0.0000	0.4954±0.0000	0.6620±0.0000	0.7060±0.0000
	GMC	0.9643±0.0000	0.9642±0.0000	0.9643±0.0000	0.9392±0.0000	0.9554±0.0000	0.9820±0.0000	0.9820±0.0000
	SM2SC	0.9683±0.0001	0.9680±0.0001	0.9685±0.0000	0.9511±0.0003	0.9604±0.0001	0.9840±0.0000	0.9840±0.0000
	LT-MSC	<b>0.9799±0.0000</b>	<b>0.9798±0.0000</b>	0.9801±0.0000	<b>0.9652±0.0000</b>	<b>0.9750±0.0000</b>	<b>0.9900±0.0000</b>	<b>0.9900±0.0000</b>
	t-SVD-MS	<b>0.9799±0.0000</b>	<b>0.9798±0.0000</b>	0.9801±0.0000	<b>0.9652±0.0000</b>	<b>0.9750±0.0000</b>	<b>0.9900±0.0000</b>	<b>0.9900±0.0000</b>
	ETLMSC	0.3699±0.0030	0.3315±0.0064	0.9730±0.0008	0.2523±0.0020	0.1881±0.0191	0.4189±0.0046	0.4525±0.0024
	CGL	0.9721±0.0000	0.9721±0.0000	0.9722±0.0000	0.9513±0.0000	0.9652±0.0000	0.9860±0.0000	0.9860±0.0000
100leaves	SC	0.5494±0.0085	0.5222±0.0096	0.5797±0.0084	0.8283±0.0032	0.5449±0.0086	0.6651±0.0081	0.6871±0.0067
	LRR	0.3572±0.0119	0.3406±0.0121	0.3755±0.0121	0.7285±0.0057	0.3508±0.0120	0.4979±0.0135	0.5294±0.0125
	MLAN	0.3626±0.0000	0.2533±0.0000	0.6378±0.0000	0.8285±0.0000	0.3539±0.0000	0.6388±0.0000	0.6694±0.0000
	MCGC	0.5637±0.0000	0.5150±0.0000	0.6787±0.0000	0.8382±0.0000	0.5592±0.0000	0.7025±0.0000	0.7262±0.0000
	GMC	0.5042±0.0000	0.3521±0.0000	0.8874±0.0000	0.9292±0.0000	0.4974±0.0000	0.8237±0.0000	0.8506±0.0000
	SM2SC	0.6461±0.0220	0.5848±0.0329	0.7330±0.0139	0.8871±0.0054	0.6423±0.0223	0.7809±0.0207	0.8057±0.0171
	LT-MSC	0.6433±0.0174	0.6129±0.0186	0.6773±0.0164	0.8699±0.0068	0.6397±0.0176	0.7339±0.0189	0.7595±0.0158
	t-SVD-MS	0.6707±0.0134	0.6388±0.0157	0.7059±0.0115	0.8829±0.0049	0.6674±0.0136	0.7542±0.0118	0.7788±0.0107
	ETLMSC	0.7164±0.0152	0.6742±0.0183	0.7670±0.0153	0.9065±0.0063	0.7135±0.0154	0.7756±0.0183	0.8001±0.0162
	CGL	<b>0.9431±0.0063</b>	<b>0.9276±0.0108</b>	<b>0.9590±0.0030</b>	<b>0.9818±0.0013</b>	<b>0.9425±0.0063</b>	<b>0.9625±0.0070</b>	<b>0.9646±0.0054</b>
COIL20	SC	0.8016±0.0005	0.7711±0.0006	0.8437±0.0102	0.9106±0.0057	0.7909±0.0006	0.8389±0.0002	0.8597±0.0002
	LRR	0.7684±0.0069	0.7394±0.0150	0.8000±0.0067	0.8688±0.0041	0.7558±0.0074	0.8010±0.0079	0.8224±0.0042
	MLAN	0.8110±0.0000	0.7213±0.0000	0.9261±0.0000	0.9405±0.0000	0.7999±0.0000	0.8424±0.0000	0.8736±0.0000
	MCGC	0.7282±0.0000	0.6808±0.0000	<b>1.0000±0.0000</b>	0.8867±0.0000	0.7130±0.0000	0.7764±0.0000	0.8139±0.0000
	GMC	0.7997±0.0000	0.6952±0.0000	0.9411±0.0000	<b>0.9415±0.0000</b>	0.7876±0.0000	0.8035±0.0000	0.8465±0.0000
	SM2SC	0.7637±0.0255	0.7028±0.0312	0.8804±0.0335	0.9077±0.0123	0.7497±0.0272	0.7684±0.0352	0.8155±0.0224
	LT-MSC	0.7183±0.0071	0.6881±0.0140	0.7528±0.0056	0.8423±0.0030	0.7030±0.0076	0.7710±0.0047	0.7840±0.0045
	t-SVD-MS	0.7273±0.0185	0.7010±0.0238	0.7559±0.0143	0.8428±0.0089	0.7126±0.0196	0.7727±0.0180	0.7882±0.0160
	ETLMSC	0.7410±0.0217	0.7311±0.0216	0.7512±0.0223	0.8422±0.0105	0.7274±0.0229	0.7788±0.0223	0.7911±0.0220
	CGL	<b>0.8440±0.0005</b>	<b>0.8238±0.0004</b>	0.8653±0.0005	0.9193±0.0000	<b>0.8357±0.0005</b>	<b>0.8596±0.0003</b>	<b>0.8832±0.0003</b>
handwritten	SC	0.9225±0.0000	0.9221±0.0000	0.9229±0.0000	0.9163±0.0000	0.9139±0.0000	0.9600±0.0000	0.9600±0.0000
	LRR	0.7290±0.0001	0.7018±0.0001	0.7585±0.0001	0.7679±0.0000	0.6978±0.0001	0.7795±0.0001	0.8105±0.0000
	MLAN	0.9475±0.0000	0.9468±0.0000	0.9482±0.0000	0.9400±0.0000	0.9417±0.0000	0.9735±0.0000	0.9735±0.0000
	MCGC	0.8970±0.0000	0.8948±0.0000	0.8991±0.0000	0.8926±0.0000	0.8856±0.0000	0.9465±0.0000	0.9465±0.0000
	GMC	0.8661±0.0000	0.8268±0.0000	0.9093±0.0000	0.9057±0.0000	0.8505±0.0000	0.8820±0.0000	0.8820±0.0000
	SM2SC	0.9252±0.0000	0.9244±0.0000	0.9259±0.0000	0.9163±0.0000	0.9169±0.0000	0.9615±0.0000	0.9615±0.0000
	LT-MSC	0.8195±0.0140	0.8167±0.0145	0.8223±0.0135	0.8346±0.0086	0.7995±0.0156	0.8982±0.0104	0.8982±0.0104
	t-SVD-MS	0.8608±0.0039	0.8577±0.0045	0.8640±0.0034	0.8653±0.0027	0.8454±0.0044	0.9255±0.0025	0.9255±0.0025
	ETLMSC	0.7629±0.0021	0.7604±0.0021	0.7654±0.0022	0.7835±0.0020	0.7366±0.0024	0.8635±0.0012	0.8635±0.0012
	CGL	<b>0.9554±0.0000</b>	<b>0.9549±0.0000</b>	<b>0.9559±0.0000</b>	<b>0.9491±0.0000</b>	<b>0.9505±0.0000</b>	<b>0.9775±0.0000</b>	<b>0.9775±0.0000</b>

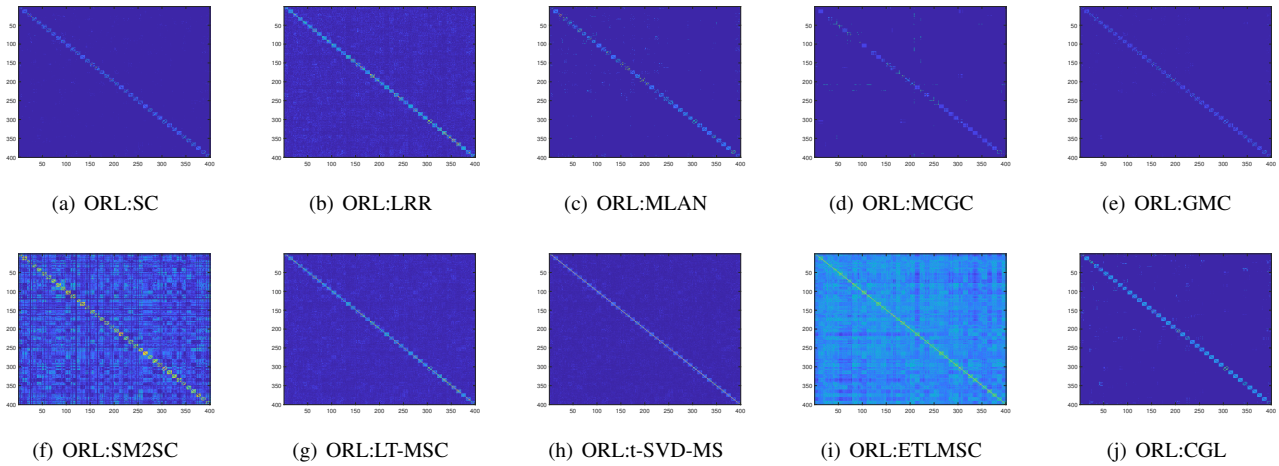


Fig. 2: Visual comparisons of the similarity matrices learned by different methods on ORL dataset.

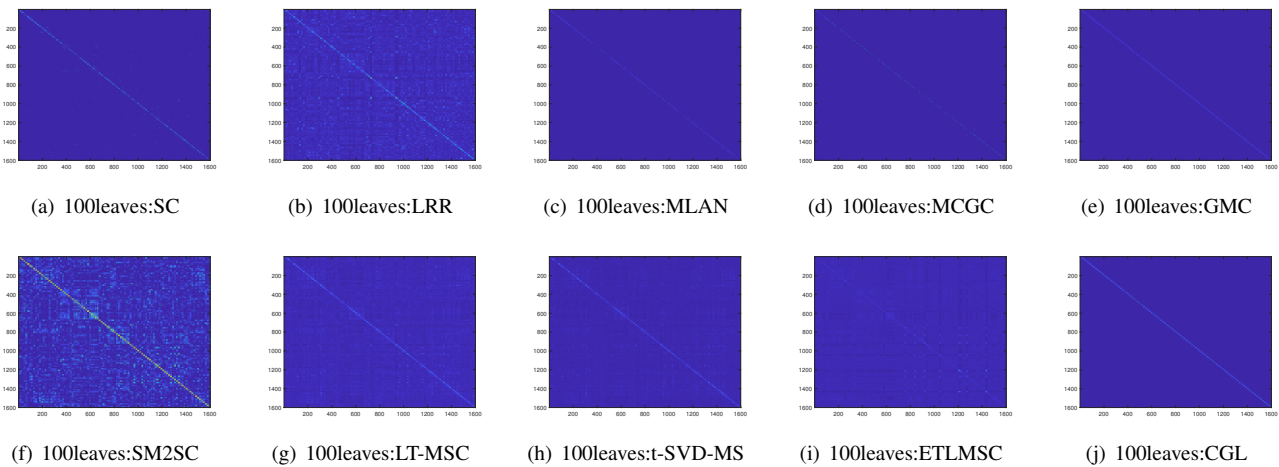


Fig. 3: Visual comparisons of the similarity matrices learned by different methods on 100leaves dataset.

(2) The proposed method performs better than three graph based multi-view clustering methods (MLAN, GMC, and ETLMSC). MLAN, GMC, and ETLMSC learn the similarity graphs on the original features, which consist of noises and redundancy information. The learned similarity graphs may be insufficient for revealing the essential cluster structures. The proposed method achieves better performance than the MCGC method, which learns the consensus similarity graph in the spectral embedding space and pairwise investigates the view correlations.

(3) The proposed method achieves better results than three tensor based multi-view clustering methods (LT-MSC, t-SVD-MS, and ETLMSC) in most cases. It indicates that constructing the similarity graph in the embedding space can achieve better performance than that in the original feature space.

(4) Compared with three subspace based multi-view clustering methods (LT-MSC, t-SVD-MS, and SM2SC), the proposed method also achieve better results in most cases. LT-MSC and t-SVD-MS perform better than the proposed method on the 20newsgroups dataset. The reason may be that the  $l_{2,1}$  norm

removals of outliers from data and subspace segmentation may favor this dataset.

(5) SC and LRR are two strong baselines, and usually obtain comparable or even better performance than some other compared methods. Nonetheless, our proposed method significantly outperforms the two baselines on all datasets, which demonstrates the superiority of our proposed method.

2) *Learned similarity graph:* We visualize the similarity graphs<sup>4</sup> learned by two single view clustering baselines, seven multi-view clustering competitors, and the proposed method on ORL and 100leaves datasets in Fig. 2 and 3. Compared with other methods, the similarity graphs of the proposed method have clearer and completer block-diagonal structures. It indicates that the similarity graphs of the proposed method possess more correct node connections and are more appropriate for the clustering tasks.

<sup>4</sup>As for the ETLMSC method, we visualize the final transition probability matrix.



3) *Learned embedded features*: In this section, we conduct one experiment to verify that the learned spectral embedding features are more beneficial to construct an intrinsic similarity graph for the clustering task than the original features. We construct view-specific and average similarity graphs on the original features and embedded ones learned in different iterations via Eq. 3 and Eq. 5, respectively. Then, we perform the spectral clustering on the similarity graphs and report the clustering performance measured by ACC in TABLE II. As can be seen, the learned embedded features can be used to construct more intrinsic similarity graphs (both view-specific and average similarity graphs) than the original ones with the iterations increasing. This strongly indicates the effectiveness of the learned embedded features.

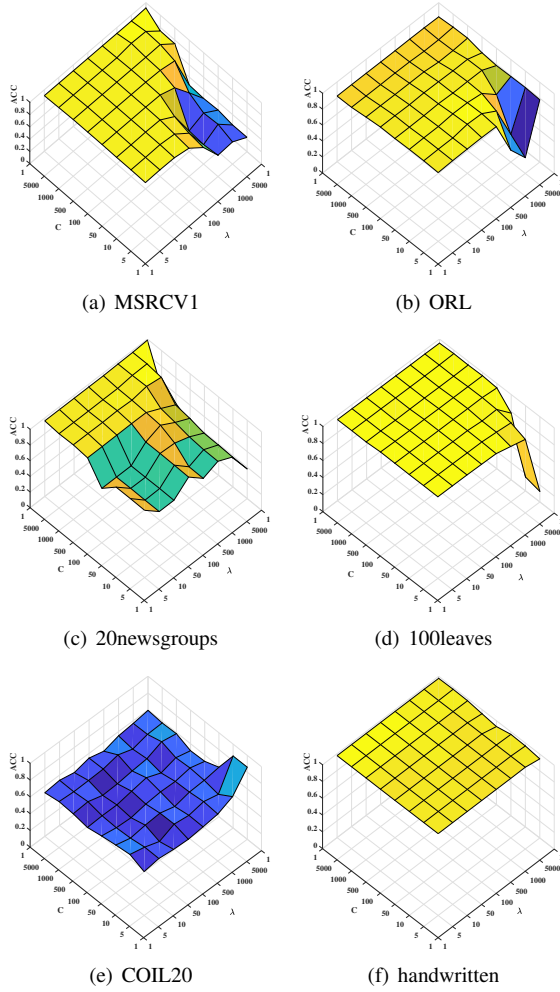


Fig. 4: Illustration of the effect of two parameters of the proposed method (i.e.,  $\lambda$  and  $C$ ) on the clustering performance in terms of ACC.

### C. Model analysis

1) *Parameter sensitivity*: The proposed method consists of three parameters, i.e.,  $k$ ,  $\lambda$  and  $C$ . In our experiments, the parameter  $k$  is set as  $k = 15$  for all datasets. Thus, we only need to tune two parameters.  $\lambda$  and  $C$  are chosen from the

same range  $[1, 5, 10, 50, 100, 500, 1000, 5000]$  with grid search strategy. Fig. 4 gives the clustering performance in terms of ACC with different combinations of parameters  $\lambda$  and  $C$  on six datasets. The proposed method can achieve stable performance on MSRCV1, ORL, 100leaves, and handwritten datasets with parameter perturbations. In contrast, the performance of the proposed method is sensitive to different parameter combinations on 20newsgrroups and COIL20 datasets.

2) *Computation complexity and empirical convergence analysis*: The computation complexity of the proposed method mainly lies in updating variables  $\{\mathbf{H}^{(v)}\}_{v=1}^V$  and  $\mathcal{T}$ . For updating  $\{\mathbf{H}^{(v)}\}_{v=1}^V$ , it takes  $\mathcal{O}(n^2c)$  to compute the  $c$  eigenvectors corresponding to the  $c$  largest eigenvalues of a  $n \times n$  matrix in each iteration for each view. As for updating  $\mathcal{T}$ , it spends  $\mathcal{O}(n^2V\log(n))$  and  $\mathcal{O}(n^2V^2)$  for the FFT/IFFT operation and conducting SVD on  $n \times V$  matrices, respectively. For solving Eq. (3) and spectral clustering, it costs  $\mathcal{O}(n\log(n))$  and  $\mathcal{O}(n^2c)$ , respectively. Thus, the total computation complexity is  $\mathcal{O}(tVn^2c + tn^2V\log(n) + tn^2V^2 + n\log(n) + n^2c)$ , where  $t$  is the number of iterations. Furthermore, we compare the time cost of the proposed method with that of other methods on the six benchmark datasets. The results of running time are reported in TABLE IV. Although the proposed method is not the most efficient one, it runs faster than some compared methods such as LT-MS, t-SVD-MS, and ETLMS on most datasets. In addition, the proposed method can obtain better results as discussed in previous sections.

To study the empirical convergence of the proposed method, we plot the convergence curves of the proposed method on six datasets in Fig. 5. We can see that the objective function values can reach a stable value within 100 iterations. Thus, the proposed method possesses a stable convergence.

3) *The effectiveness of the weighted tensor nuclear norm*: To investigate the effectiveness of the proposed weight tensor nuclear norm, we carry out two models in Eq. (7) and Eq. (9) and term them as CGL-no-LRT and CGL-TNN. For CGL-no-LRT, it directly uses the spectral embedding matrices to generate the consensus similarity graph without the procedure of low rank tensor representation learning. For CGL-TNN, it utilizes the nuclear tensor norm to capture consistency among multiple views. Two regularization parameters  $\lambda$  and  $\tau$  are also searched in the same range  $[1, 5, 10, 50, 100, 500, 1000, 5000]$ . TABLE III gives the clustering performance of CGL-no-LRT, CGL-TNN, and the proposed CGL method in terms of seven evaluation metrics on six datasets. On handwritten dataset, CGL-no-LRT, CGL-TNN, and CGL achieve similar clustering performances. On rest datasets, CGL-TNN shows better results than CGL-no-LRT and CGL. It indicates the effectiveness of the proposed weight tensor nuclear norm.

## VI. CONCLUSION

In this paper, a graph learning method is proposed to conduct clustering by simultaneously learning spectral embedding matrices and low rank tensor representation. In the proposed method, we stack multi-view gram matrices constructed by the inner product of normalized spectral matrices into a third-order tensor and employ the tensor low rank approximation to capture the global consistent information among multiple views.

TABLE II: The clustering performance measured by ACC on some intermediate representations of six different datasets.

Datasets	Similarity graphs	Original features	Embedded features					
			t=1	t=20	t=40	t=60	t=80	t=100
MSRCV1	View-specific	1	0.7548	0.7524	0.8667	0.8667	0.8714	0.8667
		2	0.2852	0.2762	0.2714	0.2714	0.2714	0.2667
		3	0.7143	0.7095	0.7381	0.7333	0.7333	0.7381
		4	0.6374	0.6236	0.8429	0.8762	0.8762	0.8667
		5	0.6183	0.6048	0.7095	0.8000	0.7952	0.8000
	Average	6	0.4205	0.4286	0.4374	0.4652	0.4588	0.4583
ORL	View-specific	1	0.8143	0.8952	0.9286	0.9381	0.9381	0.9429
		2	0.6465	0.6485	0.8764	0.8864	0.8981	0.8985
		3	0.7950	0.7949	0.8773	0.8845	0.8965	0.9006
		4	0.6943	0.6988	0.8796	0.8811	0.8859	0.8885
		5	0.6943	0.6988	0.8796	0.8811	0.8859	0.8885
	Average	6	0.7598	0.8251	0.8771	0.8821	0.8943	0.8991
20NGs	View-specific	1	0.6320	0.6300	0.9720	0.9660	0.9660	0.9700
		2	0.6600	0.6520	0.9780	0.9760	0.9740	0.9640
		3	0.3259	0.3620	0.9480	0.9200	0.9280	0.9560
		4	0.3259	0.3620	0.9480	0.9200	0.9280	0.9560
		5	0.3259	0.3620	0.9480	0.9200	0.9280	0.9560
	Average	6	0.9640	0.6300	0.9840	0.9860	0.9880	0.9860
COIL20	View-specific	1	0.7903	0.4433	0.5083	0.6589	0.6864	0.8542
		2	0.7842	0.6131	0.5694	0.5435	0.6981	0.8368
		3	0.7506	0.5896	0.5294	0.4879	0.5528	0.8667
		4	0.7506	0.5896	0.5294	0.4879	0.5528	0.8667
		5	0.7506	0.5896	0.5294	0.4879	0.5528	0.8667
	Average	6	0.8389	0.5057	0.4932	0.5009	0.6263	0.8438
100leaves	View-specific	1	0.6651	0.6730	0.7074	0.7188	0.7213	0.7252
		2	0.3694	0.3798	0.3972	0.3966	0.4040	0.4037
		3	0.5384	0.5529	0.5845	0.5912	0.5958	0.5998
		4	0.5384	0.5529	0.5845	0.5912	0.5958	0.5998
		5	0.5384	0.5529	0.5845	0.5912	0.5958	0.5998
	Average	6	0.8797	0.9503	0.9632	0.9654	0.9644	0.9681
handwritten	View-specific	1	0.9600	0.9570	0.9680	0.9730	0.9781	0.9775
		2	0.7185	0.7160	0.8140	0.8860	0.8830	0.8845
		3	0.7506	0.7485	0.9680	0.9745	0.9735	0.9775
		4	0.6638	0.6145	0.8405	0.8810	0.8350	0.8795
		5	0.9480	0.9555	0.9695	0.9735	0.9715	0.9765
	Average	6	0.4432	0.4756	0.4730	0.5450	0.5590	0.5625
		Average	0.9715	0.9760	0.9750	0.9750	0.9775	0.9785

TABLE III: Clustering performance on six 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
MSRCV1	CGL-no-LRT	0.8058±0.0000	0.7949±0.0000	0.8171±0.0000	0.8224±0.0000	0.7740±0.0000	0.8952±0.0000	0.8952±0.0000
	CGL-TNN	0.8705±0.0000	0.8668±0.0000	0.8752±0.0000	0.8683±0.0000	0.8495±0.0000	0.9333±0.0000	0.9333±0.0000
	CGL	<b>0.8945±0.0000</b>	<b>0.8914±0.0000</b>	<b>0.8975±0.0000</b>	<b>0.8883±0.0000</b>	<b>0.8774±0.0000</b>	<b>0.9476±0.0000</b>	<b>0.9476±0.0000</b>
ORL	CGL-no-LRT	0.7050±0.0175	0.6509±0.0260	0.7696±0.0106	0.8944±0.0046	0.6976±0.0181	0.7789±0.0219	0.8056±0.0161
	CGL-TNN	0.8506±0.0041	0.8363±0.0067	0.8654±0.0025	0.9418±0.0014	0.8471±0.0043	0.8918±0.0028	0.9042±0.0028
	CGL	<b>0.8584±0.0060</b>	<b>0.8446±0.0074</b>	<b>0.8727±0.0050</b>	<b>0.9454±0.0021</b>	<b>0.8551±0.0061</b>	<b>0.8996±0.0051</b>	<b>0.9074±0.0038</b>
20newsgroups	CGL-no-LRT	0.5792±0.0000	0.4698±0.0000	0.7552±0.0000	0.6358±0.0000	0.4430±0.0000	0.6300±0.0000	0.6900±0.0000
	CGL-TNN	0.9682±0.0000	0.9680±0.0000	0.9683±0.0000	0.9461±0.0000	0.9603±0.0000	0.9840±0.0000	0.9840±0.0000
	CGL	<b>0.9721±0.0000</b>	<b>0.9721±0.0000</b>	<b>0.9722±0.0000</b>	<b>0.9513±0.0000</b>	<b>0.9652±0.0000</b>	<b>0.9860±0.0000</b>	<b>0.9860±0.0000</b>
100leaves	CGL-no-LRT	0.8481±0.0163	0.7772±0.0258	0.9338±0.0046	0.9642±0.0029	0.8466±0.0165	0.8705±0.0156	0.8866±0.0136
	CGL-TNN	0.9413±0.0054	0.9264±0.0098	0.9571±0.0029	0.9809±0.0015	0.9407±0.0054	0.9623±0.0046	<b>0.9647±0.0040</b>
	CGL	<b>0.9431±0.0063</b>	<b>0.9276±0.0108</b>	<b>0.9590±0.0030</b>	<b>0.9818±0.0013</b>	<b>0.9425±0.0063</b>	<b>0.9625±0.0070</b>	0.9646±0.0054
COIL20	CGL-no-LRT	0.7952±0.0194	0.7422±0.0250	0.8567±0.0141	0.9097±0.0076	0.7838±0.0206	0.7691±0.0225	0.8056±0.0156
	CGL-TNN	0.8089±0.0019	0.7685±0.0060	0.8538±0.0071	0.9165±0.0046	0.7984±0.0020	0.8224±0.0046	0.8435±0.0058
	CGL	<b>0.8440±0.0005</b>	<b>0.8238±0.0004</b>	<b>0.8653±0.0005</b>	<b>0.9193±0.0000</b>	<b>0.8357±0.0005</b>	<b>0.8596±0.0003</b>	<b>0.8832±0.0003</b>
handwritten	CGL-no-LRT	0.9525±0.0000	0.9519±0.0000	0.9532±0.0000	0.9456±0.0000	0.9473±0.0000	0.9760±0.0000	0.9760±0.0000
	CGL-TNN	<b>0.9590±0.0000</b>	<b>0.9581±0.0000</b>	<b>0.9599±0.0000</b>	<b>0.9549±0.0000</b>	<b>0.9545±0.0000</b>	<b>0.9795±0.0000</b>	<b>0.9795±0.0000</b>
	CGL	0.9554±0.0000	0.9549±0.0000	0.9559±0.0000	0.9491±0.0000	0.9505±0.0000	0.9775±0.0000	0.9775±0.0000

We introduce a weighted tensor nuclear norm by assigning different singular values with different weights to improve the flexibility of tensor nuclear norm in low rank approximation problem. Furthermore, we unify the spectral embedding and low rank tensor learning into a unified optimization framework to achieve mutual promotion. Finally, the consensus similarity graph can be constructed from multi-view normalized spectral embedding matrices via an adaptive neighbor graph learning manner. Extensive experiments on six datasets demonstrate that the proposed method achieves state-of-the-art multi-view

clustering performance.

#### ACKNOWLEDGMENT

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

#### REFERENCES

- [1] S. Yang, L. Li, S. Wang, W. Zhang, Q. Huang, and Q. Tian, "Skeletonnet: A hybrid network with a skeleton-embedding process for multi-view image representation learning," *IEEE Transactions on Multimedia*, vol. 21, no. 11, pp. 2916–2929, 2019.

TABLE IV: The comparison of the time cost of different algorithms on the six datasets.

Datasets	SC-SB	LRR-SB	MLAN	MCGC	GMC	SM2SC	LT-MSC	t-SVD-MS	ETLMSC	CGL
MSRCV1	0.2282	2.3433	0.2733	0.6142	0.5357	0.1961	12.1080	7.8759	0.8567	4.6915
ORL	0.3760	13.4040	0.8254	4.7955	0.7919	1.7204	54.1645	36.6418	2.2368	13.2559
20newsgroups	0.2360	14.4379	0.7739	1.1311	0.8543	1.5727	44.3049	29.9509	2.8431	10.4628
100leaves	1.7064	201.6239	5.8051	57.5732	6.0110	36.6544	330.1080	151.9008	49.0698	181.7527
COIL20	1.3063	202.3926	9.2366	44.5008	8.2687	13.4238	481.4801	227.6331	48.5709	127.7634
handwritten	2.9888	736.3062	26.2652	64.4387	15.0205	25.8285	1099.0310	445.5483	157.2392	451.5799

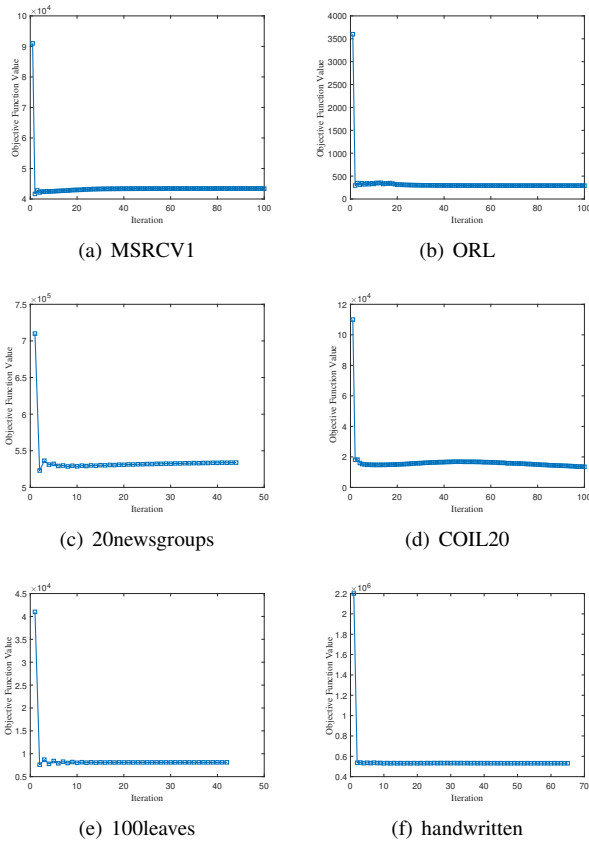


Fig. 5: The convergence curves of the proposed method on six datasets.

[2] Y. Fu, Y. Guo, Y. Zhu, F. Liu, C. Song, and Z.-H. Zhou, "Multi-view video summarization," *IEEE Transactions on Multimedia*, vol. 12, no. 7, pp. 717–729, 2010.

[3] J. Tang, X. Shu, Z. Li, Y.-G. 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.

[4] J. Tang, J. Lin, Z. Li, and J. Yang, "Discriminative deep quantization hashing for face image retrieval," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 12, pp. 6154–6162, 2018.

[5] X. Peng, Z. Huang, J. Lv, H. Zhu, and J. T. Zhou, "Comic: Multi-view clustering without parameter selection," in *International Conference on Machine Learning*, 2019, pp. 5092–5101.

[6] L. Du, P. Zhou, L. Shi, H. Wang, M. Fan, W. Wang, and Y.-D. Shen, "Robust multiple kernel k-means using l21-norm," in *International Conference on Artificial Intelligence*, 2015, pp. 3476–3482.

[7] C.-D. Wang, J.-H. Lai, and S. Y. Philip, "Multi-view clustering based on belief propagation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 4, pp. 1007–1021, 2015.

[8] Y. Chen, X. Xiao, and Y. Zhou, "Multi-view subspace clustering via simultaneously learning the representation tensor and affinity matrix," *Pattern Recognition*, vol. 106, p. 107441, 2020.

[9] B.-Y. Liu, L. Huang, C.-D. Wang, S. Fan, and S. Y. Philip, "Adaptively weighted multiview proximity learning for clustering," *IEEE Transactions on Cybernetics*, pp. 1–15, 2019.

[10] Y.-X. Ji, L. Huang, H.-P. He, C.-D. Wang, G. Xie, W. Shi, and K.-Y. Lin, "Multi-view outlier detection in deep intact space," in *IEEE International Conference on Data Mining*, 2019, pp. 1132–1137.

[11] 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*, pp. 1–1, 2020.

[12] 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*, pp. 1–1, 2020.

[13] J. Wen, K. Yan, Z. Zhang, Y. Xu, J. Wang, L. Fei, and B. Zhang, "Adaptive graph completion based incomplete multi-view clustering," *IEEE Transactions on Multimedia*, pp. 1–1, 2020.

[14] 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.

[15] H. Wang, Y. Wang, Z. Zhang, X. Fu, L. Zhuo, M. Xu, and M. Wang, "Kernelized multiview subspace analysis by self-weighted learning," *IEEE Transactions on Multimedia*, pp. 1–1, 2020.

[16] 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.

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

[18] H. Gao, F. Nie, X. Li, and H. Huang, "Multi-view subspace clustering," in *IEEE International Conference on Computer Vision*, 2015, pp. 4238–4246.

[19] X. Cao, C. Zhang, H. Fu, S. Liu, and H. Zhang, "Diversity-induced multi-view subspace clustering," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 586–594.

[20] A. Gretton, K. Fukumizu, C. H. Teo, L. Song, B. Schölkopf, and A. J. Smola, "A kernel statistical test of independence," in *Advances in Neural Information Processing Systems*, 2008, pp. 585–592.

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

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

[23] 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.

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

[25] 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.

[26] 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.

[27] 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.

[28] M. Brbić and I. Kopriva, "Multi-view low-rank sparse subspace clustering," *Pattern Recognition*, vol. 73, pp. 247–258, 2018.

[29] N. Xu, Y. Guo, X. Zheng, Q. Wang, and X. Luo, "Partial multi-view subspace clustering," in *ACM International Conference on Multimedia*, 2018, pp. 1794–1801.



- [30] 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*, 2020, pp. 4412–4419.
- [31] X. Sun, Y. Wang, and X. Zhang, "Multi-view subspace clustering via non-convex tensor rank minimization," in *IEEE International Conference on Multimedia and Expo*, 2020, pp. 1–6.
- [32] Z. Kang, X. Zhao, C. Peng, H. Zhu, J. T. Zhou, X. Peng, W. Chen, and Z. Xu, "Partition level multiview subspace clustering," *Neural Networks*, vol. 122, pp. 279–288, 2020.
- [33] C. Zhang, H. Fu, J. Wang, W. Li, X. Cao, and Q. Hu, "Tensorized multi-view subspace representation learning," *International Journal of Computer Vision*, pp. 1–18, 2020.
- [34] Z. Kang, Z. Guo, S. Huang, S. Wang, W. Chen, Y. Su, and Z. Xu, "Multiple partitions aligned clustering," in *International Joint Conference on Artificial Intelligence*, 2019, pp. 2701–2707.
- [35] Z. Kang, X. Lu, J. Yi, and Z. Xu, "Self-weighted multiple kernel learning for graph-based clustering and semi-supervised classification," in *International Joint Conference on Artificial Intelligence*, 2018, pp. 2312–2318.
- [36] Y. Xie, J. Liu, Y. Qu, D. Tao, W. Zhang, L. Dai, and L. Ma, "Robust kernelized multiview self-representation for subspace clustering," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–1, 2020.
- [37] Y. Wang, W. Zhang, L. Wu, X. Lin, M. Fang, and S. Pan, "Iterative views agreement: an iterative low-rank based structured optimization method to multi-view spectral clustering," in *International Joint Conference on Artificial Intelligence*, 2016, pp. 2153–2159.
- [38] Y. Wang, X. Lin, L. Wu, W. Zhang, Q. Zhang, and X. Huang, "Robust subspace clustering for multi-view data by exploiting correlation consensus," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3939–3949, 2015.
- [39] F. Nie, J. Li, X. Li *et al.*, "Parameter-free auto-weighted multiple graph learning: A framework for multiview clustering and semi-supervised classification," in *International Joint Conference on Artificial Intelligence*, 2016, pp. 1881–1887.
- [40] F. Nie, J. Li, and X. Li, "Self-weighted multiview clustering with multiple graphs," in *International Joint Conference on Artificial Intelligence*, 2017, pp. 2564–2570.
- [41] 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*, 2014, pp. 2149–2155.
- [42] 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.
- [43] 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, 2019.
- [44] K. Zhan, C. Niu, C. Chen, F. Nie, C. Zhang, and Y. Yang, "Graph structure fusion for multiview clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 10, pp. 1984–1993, 2019.
- [45] X. Zhu, S. Zhang, W. He, R. Hu, C. Lei, and P. Zhu, "One-step multi-view spectral clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 10, pp. 2022–2034, 2018.
- [46] Y. Liang, D. Huang, and C.-D. Wang, "Consistency meets inconsistency: A unified graph learning framework for multi-view clustering," in *IEEE International Conference on Data Mining*, 2019, pp. 1204–1209.
- [47] S. Huang, Z. Xu, I. W. Tsang, and Z. Kang, "Auto-weighted multi-view co-clustering with bipartite graphs," *Information Sciences*, vol. 512, pp. 18–30, 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 *AAAI Conference on Artificial Intelligence*, 2020, pp. 5924–5931.
- [49] 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, 2012.
- [50] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 2790–2797.
- [51] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, and S. Yan, "Tensor robust principal component analysis with a new tensor nuclear norm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 4, pp. 925–938, 2020.
- [52] F. Nie, X. Wang, and H. Huang, "Clustering and projected clustering with adaptive neighbors," in *International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 977–986.
- [53] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, and L. Zhang, "Weighted nuclear norm minimization and its applications to low level vision," *International Journal of Computer Vision*, vol. 121, no. 2, pp. 183–208, 2017.
- [54] S. A. Nene, S. K. Nayar, H. Murase *et al.*, "Columbia object image library (coil-20)," 1996.
- [55] 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.
- [56] 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.



**Zhenglai Li** received the BE degree from China University of Geosciences, Wuhan, China, in 2018. Currently, he is a graduate student in China University of Geosciences, Wuhan, China. His research interests include multi-view clustering.



**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 an associate professor at the School of Computer Science, China University of Geosciences, Wuhan, China. Dr. Tang has published 30+ 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, IEEE SPL, ICCV, CVPR, IJCAI, AAAI and ACMM, etc. He served on the Technical Program Committees of CVPR 2019/2020, ICCV 2019, ECCV 2020, NIPS 2020, IJCAI 2018/2019/2020, ICME 2018/2019/2020, AAAI 2019/2020. His current research interests include machine 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 40+ peer-reviewed papers, including those in highly regarded journals and conferences such as IEEE T-IP, IEEE T-NNLS, ICCV, AAAI, IJCAI, etc. He served on the Technical Program Committees of IJCAI 2016/2017/2018 and AAAI 2016/2017/2018.



**Xiao Zheng** received her master degree from the Tianjin Medical University, Tianjin, China. She is currently pursuing the Ph.D. degree with the National University of Defense Technology, China. Her recent research interests include machine learning and medical data processing.





**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 Supercomputer 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.