Neurocomputing 487 (2022) 75–85

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Multi-view unsupervised feature selection with tensor low-rank minimization

Haoliang Yuan [a,c,1], Junyu Li [d,1], Yong Liang [b,*], Yuan Yan Tang [e]

[a] State Key Laboratory of Quality Research in Chinese Medicines, Macau University of Science and Technology, Macau, China
[b] Peng Cheng Laboratory, Shenzhen, China
[c] Guangdong-HongKong-Macao Joint Laboratory for Smart Discrete Manufacturing, Guangzhou, China
[d] School of Computer Science & Engineering, South China University of Technology, Guangzhou, China
[e] Zhuhai UM Science & Technology Research Institute, University of Macau, Macau, China

ARTICLE INFO

Article history:
Received 10 October 2020
Revised 29 October 2021
Accepted 3 February 2022
Available online 8 February 2022
Communicated by Zidong Wang

Keywords:
Unsupervised feature selection
Tensor low-rank
Graph embedding

ABSTRACT

To describe objects more comprehensively and accurately, multi-view learning has attracted considerable attention. Recently, graph embedding based multi-view feature selection methods have been proposed and shown efficient in many real applications. The existing methods generally construct one common graph matrix to exploit the local structure of multi-view data via the linear weight fusion or learning one common graph matrix across all views. However, since all views share the identical graph structure, this emphasizes the consistency too much, resulting in restricting the diversity among different views. In this paper, a tensor low-rank constrained graph embedding method is proposed for multi-view unsupervised feature selection. To embody the view-specific information of each view, our model constructs the graph structure for each corresponding view, respectively. To capture the consistency across views, a tensor low-rank regularization constraint is imposed on the tensor data formed by these graph matrices. An efficient optimization algorithm with theoretical convergence guarantee is designed to solve the proposed method. Extensive experimental results validate that the proposed method outperforms some state-of-the-art methods. The code of our model can be found athttps://www.researchgate.net/publication/353902948_demoTLR.

张量可以用于低秩约束

1. Introduction

In many applications, the objects are often represented by different modalities or features and each of them has specific physical meaning and statistic property. For example, the remote sensing technique can use various sensors to detect a target such as the image features and the electromagnetic wave features [32]. Different visual descriptors such as SIFT [31], HOG [10] and GIST [29] can be used to represent images. Although different views reflect the same object, the feature spaces could be different among multiple views, which is known as heterogeneous features. Traditional single-view methods can only handle multi-view data by simply concatenating multi-view features as a new single feature set. However, the drawback is that it may ignore the correlation information among multiple views, which may degrade the performance or fail to work. In order to acquire more comprehensive information of an object from multi-view data, it is necessary to discover the complementary information provided by different views and the inner consistency information among all views. To process this issue, many multi-views methods [40,47,27] have been proposed and made a contribution to many applications such as bioinformatics, hyperspectral remote sensing, and data recovery [5,41,20,38,35]. To integrate the multi-view data for clustering, Chen et al. [6] propose to learn a global structure and cluster the multi-view data in embedding space, while Kang et al. [21] propose to utilize the multi-view information by fusing partitions. By leveraging the consistency and diversity simultaneously, Huang et al. [18] propose a unified framework for structured multi-view clustering. Considering the effect of cluster size, Peng et al. [30] propose a multi-view clustering method without parameter selection on cluster size, which aims to learn a space with geometric consistency and cluster assignment consistency. To explore the nonlinear relationships and consider the imbalance among different views, Huang et al. [17] propose an auto-weighted multi-view clustering method that learns multi-view similarity relationships in kernel spaces. For solving the view-insufficiency issue,

* Corresponding author at: Peng Cheng Laboratory, Shenzhen, China.
E-mail address: 2371366625@qq.com (Y. Liang).
[1] These authors have contributed equally to this work.

https://doi.org/10.1016/j.neucom.2022.02.005
0925-2312/

Huang et al. [16] propose to simultaneously recover the latent intact space from multiple insufficient views and discover the cluster structure from the intact space.

In the field of multi-view learning, multi-view data also brings the problem of a surge in the number of features including noise and redundancy. Multi-view feature selection methods, aiming to reduce the dimensionality of multiple views data and explore the correlation between views, have arisen considerable research interests in recent years [36,11,15]. Traditional graph-based multi-view unsupervised feature selection methods construct one common graph matrix to exploit the local structure of multi-view data. Currently, there exist two ways to construct the common graph structure as follows. One is to fuse multiple graphs into one common graph by linear weighted fusion, i.e., $\mathbf{S} = \sum_{v=1}^{V} \alpha_v \mathbf{S}^{(v)}$, where α_v is the weight coefficient of v-th view and $\mathbf{S}^{(v)}$ is the graph matrix of v-th view. However, in this way, the graph structure needs to be computed in advance and fixed during the embedding procedure. Hence, these pre-defined graphs are inappropriate to exploit the local structure of the multi-view features in the embedding space. The other way is to learn a common graph structure across all views, which assumes that all views share the identical graph structure $\mathbf{S}$. Nevertheless, the aim of these two ways is to yield one common graph matrix to represent the local structure of muti-view data, which emphasizes consistency too much while the diversities among different views are restricted, i.e., the strong consistency problem.

In this paper, we propose a tensor low-rank constrained graph embedding method for multi-view unsupervised feature selection, which considers both consistency and diversity compatible well. In Fig. 1, one can see that the strong consistency constraint makes all views share the same graph. If these graphs $\{\mathbf{S}, \mathbf{S}, \ldots, \mathbf{S}\}$ are stacked into a 3-order tensor $N \times N \times V$, we find that the rank of its lateral slice $(:, i, :)$ is equal to 1. First of all, lateral slide is $V \times N$ sizes and it embodies the correlations among different views and samples. Secondly, based on the observation, the rank of lateral slide is 1. It means that each view's graph provides the same information without any other complementary information, i.e., diversity. To capture the consistency and complementary information of multi-view data, we impose a low-rank constraint on lateral slides. In other words, by introducing a low-rank constraint on lateral slice instead of limiting its rank to 1, we can preserve the consistent information between views and exploit the diversity information at the same time. Therefore, we consider to respectively compute the graph matrix for its corresponding view and adopt a tensor low-rank constraint on this graph tensor formed by different graph matrices. On the one hand, our method can obtain the different graph structures for different views to preserve the diversity information. On the other hand, the tensor low-rank constraint on graph tensor can explore the high-order consistency of multiple views. Hence, our work makes consistency and diversity compatible well. Our main contributions are summarized as follows:

张量低秩约束可以探索多视图的高阶一致性

(1) To exploit the complementary information, we consider computing respectively the graph matrix for each view in the embedding space. To capture the latent consistency across views, we impose a tensor low-rank constraint on a tensor, which is stacked by these learned graph matrices. Both diversity and consistency information can be well ensured in our proposed model.

(2) An effective algorithm is presented to solve the optimization problem, together with the theoretical analyses on its convergence and computational complexity. Experimental results on several multi-view databases demonstrate the effectiveness of our proposed method and surpass some state-of-the-art competitive methods.

The paper is organized as follows. In Section 2, we briefly review and discuss the related works. In Section 3, we propose a tensor low-rank constrained graph embedding method for multi-view unsupervised feature selection, together with the theoretical analysis of convergence and computational complexity. Experiments on several multi-view data sets are conducted in Section 4. Section 5 concludes the paper.

2. Related works

Unsupervised data dimensionality reduction techniques have arisen lots of research interests and can be divided into two categories, i.e., subspace learning [19,44] and feature selection [34]. In this paper, we focus on the unsupervised feature selection methods, which are approximately divided into three groups: wrapper, filter, and embedded. Wrapper methods select the optimal feature

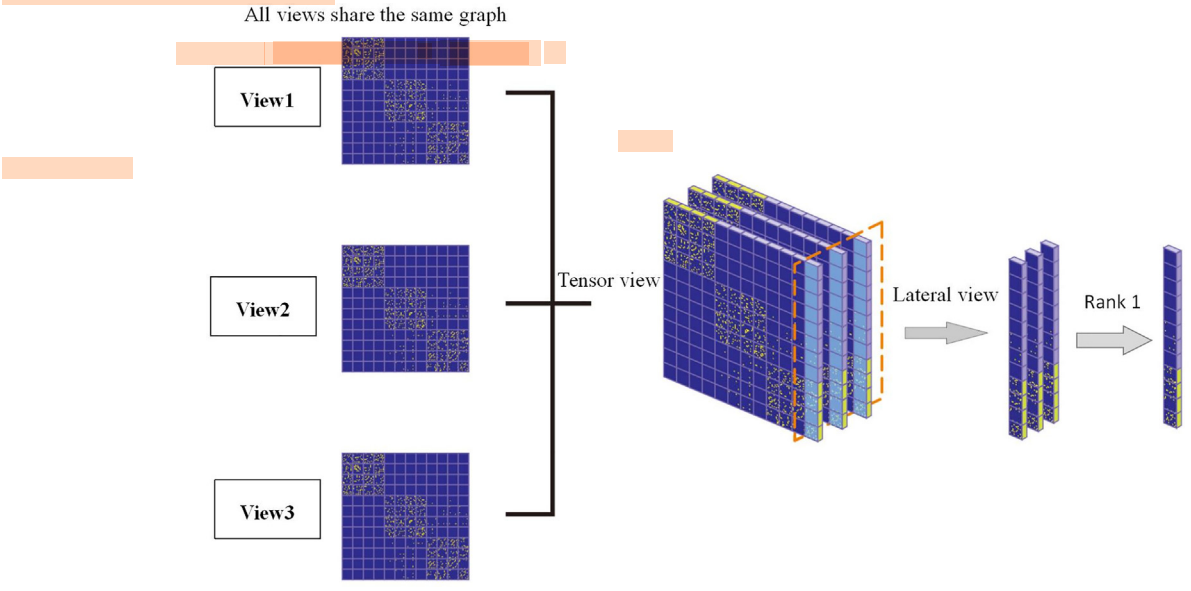

Fig. 1. An illustration of strong consistency problem caused by sharing the same graph.

张量的作用

set iteratively with respect to a predetermined feature evaluated function. The typical wrapper method is support vector machine recursive feature elimination [13]. Usually, wrapper methods are time expensive to run for the data with a large number of features. Contrarily, filter methods [8,43] are usually very efficient, since they measure the statistic characters of original data for feature selection. The representative methods include PCAScore [2], LapScor [14], SPEC [48]. Embedded methods [45,46,7,24] which embed feature selection into model construction, usually regard feature selection as a part of the learning process. The relevant features have been selected while the objective function is optimized. However, the above methods belong to the single-view model, which may be unsuitable for the multi-view data. If these single-view methods handle multi-view data by simply concatenating multi-view features as a new single feature set, they may ignore the underlying correlations between different views.

In multi-view learning, a variety of multi-view unsupervised feature selection methods have been proposed. Xu et al. [42] propose to integrate the multi-view data clustering and feature selection as a whole for processing multi-view high-dimensional data. Considering the imbalance among different views, they design two weighting schemes to assign weight for each view. AMFS [36] is an unsupervised feature selection approach proposed for human motion retrieval. AUMFS [11] is proposed for visual concept recognition. By analyzing AMFS and AUMFS, we find that the Laplacian matrices are pre-defined and fixed during the embedding. The final graph matrix is obtained by fusing these Laplacian matrices through linear weights. Hou et al. [15] propose ASVW to learn a graph matrix for exploiting the common local structures of different views. In summary, these existing methods construct one common graph matrix to ensure the consensus principle by linear weight fusion or learning from all views. Tang et al. [33] propose a CRV-DCL model, which constructs cross-view Laplacian to expand the diversity, but it abandons the consistency of graphs, which fails to make both consistency and diversity of graph compatible.

Notations and Definitions: For v-th view, we denote n vector data as $\mathbf{X}^{(v)} = \left[\mathbf{x}_1^{(v)}, \mathbf{x}_2^{(v)}, \ldots, \mathbf{x}_n^{(v)}\right] \in \mathbb{R}^{d_v \times n}$. Given a matrix $\mathbf{Z} \in \mathbb{R}^{m \times n}$, the Ł$_{2,1}$ norm is defined as $\|\mathbf{Z}\|_{2,1} = \sum_{i=1}^{m} \|\mathbf{z}^i\|_2 = \sum_{i=1}^{m} \sqrt{\sum_{j=1}^{n} Z_{ij}^2}$, where $\mathbf{z}^i = [Z_{i1}, \cdots, Z_{in}]$ denotes the i-th row of $\mathbf{Z}$, and the nuclear norm is defined as $\|\mathbf{Z}\|_* = \sum_{i=1}^{min(m,n)} |\sigma_i(\mathbf{Z})|$, where σ_i denotes the i-th largest singular value of $\mathbf{Z}$. Given a 3-order tensor $\mathcal{S} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, the tensor nuclear norm of $\mathcal{S}$ via t-SVD [39] is represented as $\|\mathcal{S}\|_{\circledast} = \|(\mathbf{F}_{n3} \otimes \mathbf{I}_{n1}) bcicr(\mathcal{S}) (\mathbf{F}_{n3}^* \otimes \mathbf{I}_{n2})\|_*$, where $\mathbf{F}_{n_3} \in \mathbb{R}^{n_3 \times n_3}$ is the Discrete Fourier Transform matrix and $\otimes$ is Kronecker product, $\mathbf{I}_n \in \mathbb{R}^{n \times n}$ is the identity matrix, and $bcicr(\mathcal{S})$ is an operation which changes 3-order tensor $\mathcal{S} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ into a block cyclic matrix as

$$bcicr(\mathcal{S}) = \begin{bmatrix} \mathbf{S}^{(1)} & \mathbf{S}^{(n_3)} & \cdots & \mathbf{S}^{(3)} & \mathbf{S}^{(2)} \\ \mathbf{S}^{(2)} & \mathbf{S}^{(1)} & \cdots & \mathbf{S}^{(4)} & \mathbf{S}^{(3)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{S}^{(n_3-1)} & \mathbf{S}^{(n_3-2)} & \cdots & \mathbf{S}^{(1)} & \mathbf{S}^{(n_3)} \\ \mathbf{S}^{(n_3)} & \mathbf{S}^{(n_3-1)} & \cdots & \mathbf{S}^{(2)} & \mathbf{S}^{(1)} \end{bmatrix}. \quad (1)$$

where $\mathbf{S}^{(i)}$ is used to represent $\mathcal{S}(:,:,i)$.

3. Proposed method

3.1. Formulation

We propose a multi-view unsupervised feature selection model by using graph embedding technique. In our model, we construct the different graph structures for multiple views in the embedding space. Besides, a tensor low-rank constraint is imposed on the tensor data, which is stacked by these graph matrices, to capture the high-order consistency across views. Hence, our model is formulated as follow:

$$\begin{aligned} &\arg \min_{\mathbf{W}^{(v)}, \mathbf{S}^{(v)}} \sum_{v=1}^{V} \sum_{i=1}^{n} \sum_{j=1}^{n} \|\mathbf{W}^{(v)T}\mathbf{x}_i^{(v)} - \mathbf{W}^{(v)T}\mathbf{x}_j^{(v)}\|^2 S_{ij}^{(v)} \\ &+\lambda_1 \sum_{v=1}^{V} \|\mathbf{W}^{(v)}\|_{2,1} + \lambda_2 \|\mathcal{S}\|_{\circledast} \\ &s.t. \quad \mathbf{W}^{(v)T}\mathbf{W}^{(v)} = \mathbf{I}, \quad \mathcal{S} = \Phi\left(\mathbf{S}^{(1)}, \mathbf{S}^{(2)}, \ldots, \mathbf{S}^{(V)}\right) \\ &\sum_{j=1}^{n} S_{ij}^{(v)} = 1, \quad S_{ij}^{(v)} \geqslant 0, \quad S_{ii}^{(v)} = 0. \end{aligned} \quad (2)$$

where the function $\Phi(\bullet)$ constructs $\left\{\mathbf{S}^{(v)}\right\}_{v=1}^{V}$ into a 3-order tensor $\mathcal{S}$ with the size of $N \times V \times N$ [39], λ_1 and λ_2 are two non-negative regularization parameters.

In model (2), the first term is used to respectively compute the graph matrix for each view under the embedding space, which aims to preserve the view-specific information for complementation. The second term is used to select the features from original multi-view data via row-sparsity. The third term is used as a tensor low-rank constraint for the tensor data, which is stacked by these graph matrices. Through model (2), we unify the graph construction, feature selection and tensor low-rank constraint as a whole. In this unified optimization procedure, the projection matrix $\mathbf{W}^{(v)} \in \mathbb{R}^{d_v \times d}$ and the graph matrix $\mathbf{S}^{(v)} \in \mathbb{R}^{n \times n}$ are obtained by the alternating procedure. With a set of orthogonal projection matrix $\mathbf{W}^{(v)}$, $v = 1, 2, \ldots, V$, tensor graph $\mathcal{S}$ seeks a solution that satisfies the tensor low-rank constraints to capture complementary and consistent information in each iteration. Then, the optimal projection matrix $\mathbf{W}^{(v)}$ can be derived from the $\mathcal{S}$.

To select a subset of representative features, we use the L_2-norm on each row of $\mathbf{W}^{(v)}$, i.e., $\|\mathbf{w}^{(v)i}\|_2$ to compute the scores of these features, where $i = 1, 2, \ldots, d_v$ and d_v is the number of features of v-th view. Then we sort these scores and choose the largest p values.

3.2. Optimization

张量低秩限制的作用是捕获互补和一致性信息，探索多视图的高阶一致性

To solve the proposed optimization problem, we develop an alternative optimizing strategy which alternatively optimizes one of the variables while the others are fixes.

• **Update $\mathbf{S}^{(v)}$**: Here, we introduce an auxiliary $\mathcal{G}$ and the optimization problem can be formulated as:

$$\begin{aligned} &\min_{\mathbf{W}^{(v)}, \mathbf{S}^{(v)}, \mathcal{G}} \sum_{v=1}^{V} \sum_{i=1}^{n} \sum_{j=1}^{n} \|\mathbf{W}^{(v)T}\mathbf{x}_i^{(v)} - \mathbf{W}^{(v)T}\mathbf{x}_j^{(v)}\|^2 S_{ij}^{(v)} \\ &+\lambda_1 \sum_{v=1}^{V} \|\mathbf{W}^{(v)}\|_{2,1} + \lambda_2 \|\mathcal{G}\|_{\circledast} \\ &s.t. \quad \mathbf{W}^{(v)T}\mathbf{W}^{(v)} = \mathbf{I}, \quad \mathcal{S} - \mathcal{G} = 0 \quad \mathcal{S} = \Phi\left(\mathbf{S}^{(1)}, \mathbf{S}^{(2)}, \ldots, \mathbf{S}^{(V)}\right) \\ &\sum_{j=1}^{n} S_{ij}^{(v)} = 1, \quad S_{ij}^{(v)} \geqslant 0, \quad S_{ii}^{(v)} = 0. \end{aligned} \quad (3)$$

Then, the above problem can be solved by Augmented Lagrange Multiplier (ALM) [25].

$$\begin{aligned} &\arg \min_{\mathbf{W}^{(v)}, \mathbf{S}^{(v)}, \mathcal{G}} \sum_{v=1}^{V} \sum_{i=1}^{n} \sum_{j=1}^{n} \|\mathbf{W}^{(v)T}\mathbf{x}_i^{(v)} - \mathbf{W}^{(v)T}\mathbf{x}_j^{(v)}\|^2 S_{ij}^{(v)} \\ &+\lambda_1 \sum_{v=1}^{V} \|\mathbf{W}^{(v)}\|_{2,1} + \lambda_2 \|\mathcal{G}\|_{\circledast} + \langle \mathcal{R}, \mathcal{S} - \mathcal{G} \rangle + \frac{\mu}{2} \|\mathcal{S} - \mathcal{G}\|_F^2 \\ &s.t. \quad \mathbf{W}^{(v)T}\mathbf{W}^{(v)} = \mathbf{I}, \quad \mathcal{S} = \Phi\left(\mathbf{S}^{(1)}, \mathbf{S}^{(2)}, \ldots, \mathbf{S}^{(V)}\right) \\ &\sum_{j=1}^{n} S_{ij}^{(v)} = 1, \quad S_{ij}^{(v)} \geqslant 0, \quad S_{ii}^{(v)} = 0. \end{aligned} \quad (4)$$

where $\mathcal{R}$ is the Lagrange multipliers and μ is the penalty parameter. To solve it, we give some equations as follows,

$$\langle \mathcal{R}, \mathcal{S}-\mathcal{G}\rangle = \sum_{v=1}^{V} tr\left(\mathbf{R}^{(v)T}\left(\mathbf{S}^{(v)}-\mathbf{G}^{(v)}\right)\right), \tag{5}$$

$$\|\mathcal{S}-\mathcal{G}\|_F^2 = \sum_{v=1}^{V}\|\mathbf{S}^{(v)}-\mathbf{G}^{(v)}\|_F^2, \tag{6}$$

where $\mathbf{S}^{(v)}=\Phi_{(v)}^{-1}(\mathcal{S})$ and $\mathbf{G}^{(v)}=\Phi_{(v)}^{-1}(\mathcal{G})$. When $\mathbf{W}^{(v)}$ and $\mathcal{G}$ are fixed, we have V subproblems for $S_{ij}^{(v)}$:

$$\min_{S_{ij}^{(v)}}\|S_{ij}^{(v)}-\frac{\mu Z_{ij}^{(v)}-\beta_{ij}^{(v)}}{\mu}\|^2, s.t. \sum_{j=1}^{n} S_{ij}^{(v)}=1, S_{ij}^{(v)}\geqslant 0, S_{ii}^{(v)}=0. \tag{7}$$

where $\beta_{ij}^{(v)}=\|\mathbf{W}^{(v)T}\mathbf{x}_i^{(v)}-\mathbf{W}^{(v)T}\mathbf{x}_j^{(v)}\|^2$ and $Z_{ij}^{(v)}=G_{ij}^{(v)}-\frac{R_{ij}^{(v)}}{\mu}$. We use Ω_{-i} to denote a set which does not contain i-th element. Due to $\mathbf{s}_{\Omega_{-i}}^{(v)T}\mathbf{e}+S_{ii}^{(v)}=1$ and $S_{ii}^{(v)}=0$, we have $\mathbf{s}_{\Omega_{-i}}^{(v)T}\mathbf{e}=1$, where $\mathbf{s}_{\Omega_{-i}}^{(v)T}$ denotes the i-th column except for i-th element of $\mathbf{S}^{(v)}$ and $\mathbf{e}=[1,1,\ldots,1]^T$. Hence, the Lagrangian function of model (7) can be reformulated as

$$\frac{1}{2}\|\mathbf{s}_{\Omega_{-i}}^{(v)}-\mathbf{u}_{\Omega_{-i}}^{(v)}\|^2-\gamma^{(v)}\left(\mathbf{s}_{\Omega_{-i}}^{(v)T}\mathbf{e}-1\right)-\lambda_{\Omega_{-i}}^{(v)T}\mathbf{s}_{\Omega_{-i}}^{(v)}, \tag{8}$$

where $\mathbf{u}_{\Omega_{-i}}^{(v)}=\frac{\mu\mathbf{Z}_{\Omega_{-i}}^{(v)}-\beta_{\Omega_{-i}}^{(v)}}{\mu}$, $\gamma^{(v)}$ is a scalar and $\lambda_{\Omega_{-i}}^{(v)}$ is a Lagrangian coefficient vector.

Then according to the KKT condition [3] and $\mathbf{s}_{\Omega_{-i}}^{(v)T}\mathbf{e}=1$, we have

$$\begin{cases}\gamma^{(v)*}=\frac{1-\mathbf{e}^T\mathbf{u}_{\Omega_{-i}}^{(v)}-\mathbf{e}^T\lambda_{\Omega_{-i}}^{(v)*}}{n}\\ S_{ij}^{(v)*}=\left(M_{ij}^{(v)}-\bar{\lambda}_i^{(v)*}\right)_+ & ,i\neq j\\ S_{ii}=0 & ,j=i\end{cases} \tag{9}$$

where $\mathbf{m}_{\Omega_{-i}}^{(v)}=\mathbf{u}_{\Omega_{-i}}^{(v)}-\frac{\mathbf{e}\mathbf{e}^T}{n}\mathbf{u}_{\Omega_{-i}}^{(v)}+\frac{1}{n}\mathbf{e}$ and $\bar{\lambda}_i^{(v)*}=\frac{\mathbf{e}^T\lambda_{\Omega_{-i}}^{(v)*}}{n}$. $M_{ij}^{(v)}$ is the j-th elements with respect to $\mathbf{m}_{\Omega_{-i}}^{(v)}$.

Algorithm 1: Optimization for $\mathbf{s}_i^{(v)}$

Optimization for $\mathbf{s}_i^{(v)}$
Input: $\mathbf{u}_{\Omega_{-i}}^{(v)}$, ε.
Output: $\mathbf{s}_i^{(v)}$.
1: Let $\mathbf{m}_{\Omega_{-i}}^{(v)}=\mathbf{u}_{\Omega_{-i}}^{(v)}-\frac{\mathbf{e}\mathbf{e}^T}{n}\mathbf{u}_{\Omega_{-i}}^{(v)}+\frac{1}{n}\mathbf{e}$, $\bar{\lambda}_i^{(v)}=0$.
2: **while** $\mathbf{s}_{\Omega_{-i}}^{(v)T}\mathbf{e}-1\geqslant\varepsilon$. **do**
3: **for** $v=1:V$ **do**
4: **for** $i=1:N$ **do**
5: $\mathbf{s}_{\Omega_{-i}}^{(v)}=\left(\mathbf{m}_{\Omega_{-i}}^{(v)}-\bar{\lambda}_i^{(v)}\right)_+$, $\bar{\lambda}_i^{(v)*}=\bar{\lambda}_i^{(v)}-\frac{f\left(\bar{\lambda}_i^{(v)}\right)}{f\prime\left(\bar{\lambda}_i^{(v)}\right)}$,
6: $S_{ii}^{(v)}=0$.
7: **end for**
8: **end for**
9: **end while**
10: **return** $\mathbf{s}_i^{(v)}$.

Eq. (9) shows that if $\bar{\lambda}_i^{(v)*}$ is fixed, we can obtain the optimal $\mathbf{s}_i^{(v)*}$. Similarly, we review the KKT condition, and get $\lambda_{ij}^{(v)*}=\left(\bar{\lambda}_i^{(v)*}-M_{ij}^{(v)}\right)_+$. Therefore, we have $\bar{\lambda}_i^{(v)}=\frac{1}{n}\sum_{j=1}^{n}\left(\bar{\lambda}_i^{(v)}-M_{ij}^{(v)}\right)_+$. We can define a function and find its root to update the $\bar{\lambda}_i^{(v)}$ by Newton method,

$$\begin{cases} f\left(\bar{\lambda}_i^{(v)}\right)=\frac{1}{n}\sum_{j=1}^{n}\left(\bar{\lambda}_i^{(v)}-M_{ij}^{(v)}\right)_+-\bar{\lambda}_i^{(v)}\\ \bar{\lambda}_{i(t+1)}^{(v)}=\bar{\lambda}_{i(t)}^{(v)}-\frac{f\left(\bar{\lambda}_{i(t)}^{(v)}\right)}{f\prime\left(\bar{\lambda}_{i(t)}^{(v)}\right)}\end{cases} \tag{10}$$

The overall procedure to compute $\mathbf{s}_i^{(v)}$ is described in Algorithm 1.

• **Update** $\mathbf{W}^{(v)}$: When $\mathbf{S}^{(v)}$ and $\mathcal{G}$ are fixed, we can solve the following function to update $\mathbf{W}^{(v)}$,

$$\begin{aligned}&\min_{\mathbf{W}^{(v)}}\sum_{v=1}^{V}\sum_{i=1}^{n}\sum_{j=1}^{n}\|\mathbf{W}^{(v)T}\mathbf{x}_i^{(v)}-\mathbf{W}^{(v)T}\mathbf{x}_j^{(v)}\|^2 S_{ij}^{(v)}\\ &+\lambda_1\sum_{v=1}^{V}\|\mathbf{W}^{(v)}\|_{2,1},\quad s.t.\quad \mathbf{W}^{(v)T}\mathbf{W}^{(v)}=\mathbf{I}.\end{aligned} \tag{11}$$

Recalling the definition of $L_{2,1}$-norm, given $\mathbf{A}\in\mathbb{R}^{d\times n}$, we have $\|\mathbf{A}\|_{2,1}=\sum_{i=1}^{d}\|\mathbf{A}^i\|_2$. We can take the derivative of $\|\mathbf{A}\|_{2,1}$ with respect to $\mathbf{A}$, $\frac{\partial\|\mathbf{A}\|_{2,1}}{\partial\mathbf{A}}=2\mathbf{O}\mathbf{A}$, where $\mathbf{O}$ is a diagonal matrix with i-th diagonal element $O_{ii}=\frac{1}{2\|\mathbf{A}^i\|_2}$. It is also worth noting that $\frac{\partial tr\left(\mathbf{A}^T\mathbf{O}\mathbf{A}\right)}{\partial\mathbf{A}}=2\mathbf{O}\mathbf{A}$. Thereby, Eq. (11) can be regarded as the following subproblem

$$\min_{\mathbf{W}^{(v)}} tr\left(\mathbf{W}^{(v)T}\mathbf{P}^{(v)}\mathbf{W}^{(v)}\right)\quad s.t.\quad \mathbf{W}^{(v)T}\mathbf{W}^{(v)}=\mathbf{I},$$

where $\mathbf{P}^{(v)}=\mathbf{X}^{(v)}\mathbf{L}^{(v)}\mathbf{X}^{(v)T}+\lambda_1\mathbf{O}^{(v)}$, $\mathbf{O}^{(v)}$ is a diagonal matrix and $O_{ii}^{(v)}=\frac{1}{2\|\mathbf{w}^{(v)i}\|_2}$. $\mathbf{L}^{(v)}$ is the Laplacian matrix derived from the symmetric matrix $\frac{\mathbf{S}^{(v)}+\mathbf{S}^{(v)}}{2}$. Therefore, the v-th optimization subproblem in model (12) can be solved by eigen-decomposition of $\mathbf{P}^{(v)}$. The column vectors of $\mathbf{W}^{(v)}\in\mathbb{R}^{d_v\times d}$ are formed by the eigenvectors with respect to the smallest d eigenvalues.

• **Update** $\mathcal{G}$: When $\mathbf{W}^{(v)}$ and $\mathcal{S}$ are obtained, we solve $\mathcal{G}$ as:

$$\min_{\mathcal{G}}\lambda_2\|\mathcal{G}\|_{\circledast}+\frac{\mu}{2}\|\mathcal{G}-\left(\mathcal{S}+\frac{\mathcal{R}}{\mu}\right)\|_F^2. \tag{13}$$

We transform model (13) into Fourier domain:

$$\begin{aligned}&\lambda_2\|bdiag\left(\mathcal{G}_f^{(1)},\ldots,\mathcal{G}_f^{(n)}\right)\|_*+\frac{\mu}{2n}\|\mathcal{G}_f-\left(\mathcal{S}_f+\frac{\mathcal{R}_f}{\mu}\right)\|_F^2\\ &\Longleftrightarrow\sum_{i=1}^{n}\frac{n\lambda_2}{\mu}\|\mathcal{G}_f^{(i)}\|_*+\frac{1}{2}\|\mathcal{G}_f^{(i)}-\left(\mathcal{S}_f^{(i)}+\frac{\mathcal{R}_f^{(i)}}{\mu}\right)\|_F^2,\end{aligned} \tag{14}$$

where $\mathcal{G}_f=fft(\mathcal{G},[],3)$ in MATLAB definition. Then we separate model (14) into n subproblems:

$$\min_{\mathcal{G}_f^{(i)}}\quad \tau\|\mathcal{G}_f^{(i)}\|_*+\frac{1}{2}\|\mathcal{G}_f^{(i)}-\left(\mathcal{S}_f^{(i)}+\frac{\mathcal{R}_f^{(i)}}{\mu}\right)\|_F^2, \tag{15}$$

where $\tau=\frac{n\lambda_2}{\mu}$. It is the F-norm based nuclear norm approximation problem in Fourier domain, which can be solved by a soft-thresholding operation [4],

$$\mathcal{G}_f^{(i)}=\mathbf{D}_\tau\left(\mathcal{S}_f^{(i)}+\frac{\mathcal{R}_f^{(i)}}{\mu}\right)=\mathcal{U}_f^{(i)}\sigma_{f,\tau}^{(i)}\mathcal{V}_f^{(i)T}, \tag{16}$$

where $\mathbf{D}_\tau(\bullet)$ is the SVT operation with threshold τ and $\sigma_{f,\tau}^{(i)}=diag\left\{\left(\sigma_f^{(i)}(j,j)-\tau\right)_+\right\}|_{j=min\left(size\left(U_f^{(i)}\right),size\left(V_f^{(i)}\right)\right)}$.

Then we have

$$\begin{cases} \mathcal{G}_f = blockfold\Big\{bdiag(\mathcal{U}_f)bdiag(\sigma_{f,\tau})bdiag\Big(\mathcal{V}_f^T\Big)\Big\} \\ \mathcal{G} = ifft(\mathcal{G}_f,[],3) \end{cases} \tag{17}$$

Hence, the overall procedure to solve our proposed method is described in Algorithm 2.

Algorithm 2: Optimization for our model.

Input: Multi-view data set $\left\{\mathbf{X}^{(1)} \in \mathbb{R}^{d_1 \times n}, \ldots, \mathbf{X}^{(V)} \in \mathbb{R}^{d_v \times n}\right\}$.
Parameter: λ_1 and λ_2, $\rho > 0$, $\mu > 0$.
Output: $\left\{\mathbf{W}^{(v)}\right\}_{v=1}^{V}$.
1: Initialize $\mathbf{R}^{(v)} = \mathbf{G}^{(v)} = \mathbf{O}_n$, $\mathbf{O}^{(v)} = \mathbf{I}_{d_v}$, $\mathbf{S}^{(v)}$ by Laplacian graph [1].
2: **while** $\|\mathbf{S}^{(v)} - \mathbf{G}^{(v)}\|_\infty \geqslant \varepsilon$. **do**
3: **for** $v = 1 : V$ **do**
4: Updating $\mathbf{W}^{(v)}$ by solving model (12) with eigen-decomposition and $O_{ii}^{(v)} = \frac{1}{2\|\mathbf{w}^{(v)i}\|_2}$.
5: Updating $\mathbf{S}^{(v)}$ by using Algorithm 1.
6: **end for**
7: Constructing tensor $\mathcal{S}$ by $\left\{\mathbf{S}^{(v)}\right\}_{v=1}^{V}$.
8: Updating $\mathcal{G}_f$ in model (14) with SVT operator and $\mathcal{G} = ifft(\mathcal{G}_f,[],3)$.
9: Updating $\mathcal{R}$ according to $\mathcal{R} \leftarrow \mathcal{R} + \mu(\mathcal{S} - \mathcal{G})$.
10: Updating μ according to $\mu \leftarrow \rho\mu$.
11: **end while**
12: **return** $\left\{\mathbf{W}^{(v)}\right\}_{v=1}^{V}$.

3.3. Convergence behavior

In this section, we provide a proposition that the objective function value of our proposed method is non-increasing by employing Algorithm 2. Before going to details, we introduce the following lemma,

Lemma 1 [28]: For any non-zero vectors $\mathbf{b}, \mathbf{d} \in \mathbb{R}^m$, the following inequality holds,

$$\|\mathbf{b}\|_2 - \frac{\|\mathbf{b}\|_2^2}{2\|\mathbf{d}\|_2} \leqslant \|\mathbf{d}\|_2 - \frac{\|\mathbf{d}\|_2^2}{2\|\mathbf{d}\|_2}. \tag{18}$$

Proposition 1: The objective function in model (4) is non-increasing by employing Algorithm 2.

Proof: We denote $\left\{\mathbf{S}_t^{(v)}\right\}_{v=1}^{V}, \left\{\mathbf{W}_t^{(v)}\right\}_{v=1}^{V}, \mathcal{G}_t$ for t-th iteration. When $\left\{\mathbf{S}^{(v)}\right\}_{v=1}^{V}$ and $\mathcal{G}$ are fixed, the optimization of model (12) is the eigen-decomposition problem, we hold

$$\begin{aligned} &tr\Big(\mathbf{W}_{t+1}^{(v)T}\mathbf{X}^{(v)}\mathbf{L}^{(v)}\mathbf{X}^{(v)T}\mathbf{W}_{t+1}^{(v)}\Big) + \lambda_1 \sum_{j=1}^{d_v} \frac{\|\mathbf{w}_{t+1}^{(v)j}\|_2^2}{2\|\mathbf{w}_t^{(v)j}\|_2} \\ &\leqslant tr\Big(\mathbf{W}_t^{(v)T}\mathbf{X}^{(v)}\mathbf{L}^{(v)}\mathbf{X}^{(v)T}\mathbf{W}_t^{(v)}\Big) + \lambda_1 \sum_{j=1}^{d_v} \frac{\|\mathbf{w}_t^{(v)j}\|_2^2}{2\|\mathbf{w}_t^{(v)j}\|_2}, \end{aligned} \tag{19}$$

where $\mathbf{w}^{(v)j}$ is the j-th row of $\mathbf{W}^{(v)}$. According to $\|\mathbf{W}^{(v)}\|_{2,1} = \sum_{j=1}^{d_v} \|\mathbf{w}^{(v)j}\|_2$, the inequality holds

$$\begin{aligned} &tr\Big(\mathbf{W}_{t+1}^{(v)T}\mathbf{X}^{(v)}\mathbf{L}^{(v)}\mathbf{X}^{(v)T}\mathbf{W}_{t+1}^{(v)}\Big) + \lambda_1\|\mathbf{W}_{t+1}^{(v)}\|_{2,1} + \lambda_1 \sum_{j=1}^{d_v}\Big(\frac{\|\mathbf{w}_{t+1}^{(v)j}\|_2^2}{2\|\mathbf{w}_t^{(v)j}\|_2} - \|\mathbf{w}_{t+1}^{(v)j}\|_2\Big) \leqslant \\ &tr\Big(\mathbf{W}_t^{(v)T}\mathbf{X}^{(v)}\mathbf{L}^{(v)}\mathbf{X}^{(v)T}\mathbf{W}_t^{(v)}\Big) + \lambda_1\|\mathbf{W}_t^{(v)}\|_{2,1} + \lambda_1 \sum_{j=1}^{d_v}\Big(\frac{\|\mathbf{w}_t^{(v)j}\|_2^2}{2\|\mathbf{w}_t^{(v)j}\|_2} - \|\mathbf{w}_t^{(v)j}\|_2\Big). \end{aligned} \tag{20}$$

Recalling the results in Lemma 1, we know $\frac{\|\mathbf{w}_{t+1}^{(v)j}\|_2^2}{2\|\mathbf{w}_t^{(v)j}\|_2} - \|\mathbf{w}_{t+1}^{(v)j}\|_2 \geqslant \frac{\|\mathbf{w}_t^{(v)j}\|_2^2}{2\|\mathbf{w}_t^{(v)j}\|_2} - \|\mathbf{w}_t^{(v)j}\|_2$. and get:

$$\begin{aligned} &tr\Big(\mathbf{W}_{t+1}^{(v)T}\mathbf{X}^{(v)}\mathbf{L}^{(v)}\mathbf{X}^{(v)T}\mathbf{W}_{t+1}^{(v)}\Big) + \lambda_1\|\mathbf{W}_{t+1}^{(v)}\|_{2,1} \\ &\leqslant tr\Big(\mathbf{W}_t^{(v)T}\mathbf{X}^{(v)}\mathbf{L}^{(v)}\mathbf{X}^{(v)T}\mathbf{W}_t^{(v)}\Big) + \lambda_1\|\mathbf{W}_t^{(v)}\|_{2,1}. \end{aligned} \tag{21}$$

Therefore, we have the following inequality and the optimization algorithm of $\mathbf{W}^{(v)}$ is convergent.

$$L\Big(\left\{\mathbf{S}_t^{(v)}\right\}_{v=1}^{V}, \left\{\mathbf{W}_{t+1}^{(v)}\right\}_{v=1}^{V}, \mathcal{G}_t\Big) \leqslant L\Big(\left\{\mathbf{S}_t^{(v)}\right\}_{v=1}^{V}, \left\{\mathbf{W}_t^{(v)}\right\}_{v=1}^{V}, \mathcal{G}_t\Big). \tag{22}$$

When $\left\{\mathbf{W}^{(v)}\right\}_{v=1}^{V}$ and $\mathcal{G}$ are fixed, we update $\left\{\mathbf{S}^{(v)}\right\}_{v=1}^{V}$ by Algorithm 1. Since model (7) is a convex optimization problem and we solve it by using KKT condition, we can get the optimal solution of $\left\{\mathbf{S}^{(v)}\right\}_{v=1}^{V}$. Therefore, we have the following inequality:

$$L\Big(\left\{\mathbf{S}_{t+1}^{(v)}\right\}_{v=1}^{V}, \left\{\mathbf{W}_{t+1}^{(v)}\right\}_{v=1}^{V}, \mathcal{G}_t\Big) \leqslant L\Big(\left\{\mathbf{S}_t^{(v)}\right\}_{v=1}^{V}, \left\{\mathbf{W}_{t+1}^{(v)}\right\}_{v=1}^{V}, \mathcal{G}_t\Big). \tag{23}$$

When we get the optimal $\left\{\mathbf{S}^{(i)}\right\}_{i=1}^{V}, \left\{\mathbf{W}^{(i)}\right\}_{i=1}^{V}$ and fix them, we can solve the minimized optimization problem in model (15) and get the optimal solution of $\mathcal{G}$ by using tensor tubal-shrinkage operator, which has been guaranteed in [39]. So we have the following inequality:

$$\begin{aligned} &L\Big(\left\{\mathbf{S}_{t+1}^{(v)}\right\}_{v=1}^{V}, \left\{\mathbf{W}_{t+1}^{(v)}\right\}_{v=1}^{V}, \mathcal{G}_{t+1}\Big) \\ &\leqslant L\Big(\left\{\mathbf{S}_{t+1}^{(v)}\right\}_{v=1}^{V}, \left\{\mathbf{W}_{t+1}^{(v)}\right\}_{v=1}^{V}, \mathcal{G}_t\Big). \end{aligned} \tag{24}$$

By combining three inequality (22), (23) and (24), the objective function value of model (2) is non-increasing:

$$\begin{aligned} &L\Big(\left\{\mathbf{S}_{t+1}^{(v)}\right\}_{v=1}^{V}, \left\{\mathbf{W}_{t+1}^{(v)}\right\}_{v=1}^{V}, \mathcal{G}_{t+1}\Big) \\ &\leqslant L\Big(\left\{\mathbf{S}_t^{(v)}\right\}_{v=1}^{V}, \left\{\mathbf{W}_t^{(v)}\right\}_{v=1}^{V}, \mathcal{G}_t\Big). \end{aligned} \tag{25}$$

3.4. Computational complexity

We analyze the computational complexity of Algorithm 2 at each iteration. O notation is to denote the time complexity. The main complexity of our algorithm lies in updating $\left\{\mathbf{S}^{(v)}\right\}_{v=1}^{V}, \left\{\mathbf{W}^{(v)}\right\}_{v=1}^{V}$ and $\mathcal{G}$. (1) We solve $\left\{\mathbf{S}^{(v)}\right\}_{v=1}^{V}$ in Newton method. For each of $\mathbf{S}^{(v)}$, its computational complexity is $O(nlog(n))$. So computational complexity of $\left\{\mathbf{S}^{(v)}\right\}_{v=1}^{V}$ is $O(nVlog(n))$. (2) We update $\left\{\mathbf{W}^{(v)}\right\}_{v=1}^{V}$ by eigen-decomposition. For each of $\mathbf{w}^{(v)}$, its computational complexity is $O\Big(d_v^3\Big)$. Therefore, the computational complexity to all views is $O\Big(\sum_{v=1}^{V} d_v^3\Big)$. When we obtain the $\mathbf{W}^{(v)}$, each $O_{ii}^{(v)}$ is corresponding to $\|\mathbf{w}^{(v)i}\|_2$. So the computational complexity is $O\Big(\sum_{v=1}^{V} d_v \times d\Big)$, where c is the number of class. Since $d_v^2 \gg d$, the total computational complexity of $\left\{\mathbf{W}^{(v)}\right\}_{v=1}^{V}$ is $O\Big(\sum_{v=1}^{V} d_v^3\Big)$. (3) As for $\mathcal{G}$, we calculate the $n \times V \times n$ tensor FFT, inverse FFT, and n SVD of $n \times V$ matrix. The first part

will take $O(2n^2Vlog(n))$. The computational cost of SVD for n matrices with size of $n \times V$ is $O\left(n^2V^2\right)$. Since in multi-view setting we have $log(n) > V$. So the computational complexity of $\mathcal{G}$ is approximate to $O(2n^2Vlog(n))$. Overall, total computational complexity of our method is approximate to $O\left((2n+1)nVlog(n) + \sum_{v=1}^{V} d_v^3\right)$.

4. Experiments

4.1. Data sets and experiment setup

Before conducting experiments, we first give the details of six databases in Table 1. We employ several competing unsupervised feature selection methods to evaluate the performance of our method, including PCAScore [2], SPEC [48], LapScor [14], NDFS [24], WMCFS [42], AUMFS [11], ASVW [15], CRV-DCL [33]. To examine the selected features are effective or not, we use the clustering result of all features as the baseline. In the clustering experiments, unsupervised feature selection methods give the scores of all features and rank them by descending order. We adopt the K-means algorithm for clustering. Clustering accuracy (ACC) and normalized mutual information (NMI) are used to evaluate the performances of clustering. We repeat the clustering at 50 times and report the average results with standard deviation (std). The parameters in all algorithms are tuned within a range set. λ_1 and λ_2 in our model are tuned from $\left\{10^{-3}, \ldots, 10^2, 10^3\right\}$ and $\{0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1\}$ respectively. d in $\mathbf{W}^v$ is a positive integer which can refer to an estimate of the number of classes. We show the parameters setting experiments in Fig. 2 and the clustering results vary with the different parameters. In order to eliminate the magnitude difference in multi-view data, we normalize data for each view by zero-mean normalization.

Table 1
Brief description of different databases.

Database	Number of Instances	Classes	Views (Dimensions)
MSRC-v1 [37]	210	7	LBP (1302) HOG (48) GIST (512) CMT (100) CENTRIST (256) SIFT (210)
BBCSport [12]	544	5	view1(3183) view2(3203)
3Sources [26]	169	6	BBC (3560) Reuters (3631) The Guardian (3068)
Digitour	1000	10	dig1-10 (64) Mnist (784) USPS (256)
Texture25 [9]	1000	25	GIST (20) PHOG (59) LBP (40)
Scene-15 [22]	4485	15	PHOW (20) LBP (59) CENTRIST(40)
ANIMAL [23]	10158	50	deep feature 1 (4096) deep feature 2 (4083)

4.2. Clustering results

For showing the performance of different unsupervised feature selection methods on the different number of selected features, we illustrate the clustering accuracy results in Fig. 3. The results show that the clustering results of different methods vary with the number of selected features. Overall, our proposed method has good performance within a range of selected features, especially in the BBCSport database, Digital database and Scene-15 database, since our proposed method not only achieves the highest clustering accuracy, but also outperforms other comparative methods in more than half of feature dimensions. In Table 2 and Table 3, we show the experimental results about the ACC and NMI of different methods under the optimal dimension and average dimension (Here, we use the average value of these different dimensions). We sum up the clustering results in Tables 2,3 and Figs. 3,4 and have following observations. (1) Multi-view feature selection is important and efficient, especially for high-dimensional data. For example, in Digitour database, our method achieves 73.11% clustering accuracy with only 56 features while baseline is 54.46% with using all features. Feature selection not only reduces the dimensionality of data but also finds a subset of discriminating and representative features to improve performance. (2) Multi-view feature selection methods take the correlations among multiple views into consideration and usually perform better or equally compared with baseline. On the contrary, single-view feature selection methods are usually inferior or approximate to baseline because they simply concatenate multi-view features as a new single feature set. For example, in Table 2, SPEC fails to outperform baseline in five databases and LapScor fails in four databases, but multi-view methods AUMFS, ASVW, CRV-DCL and our proposed

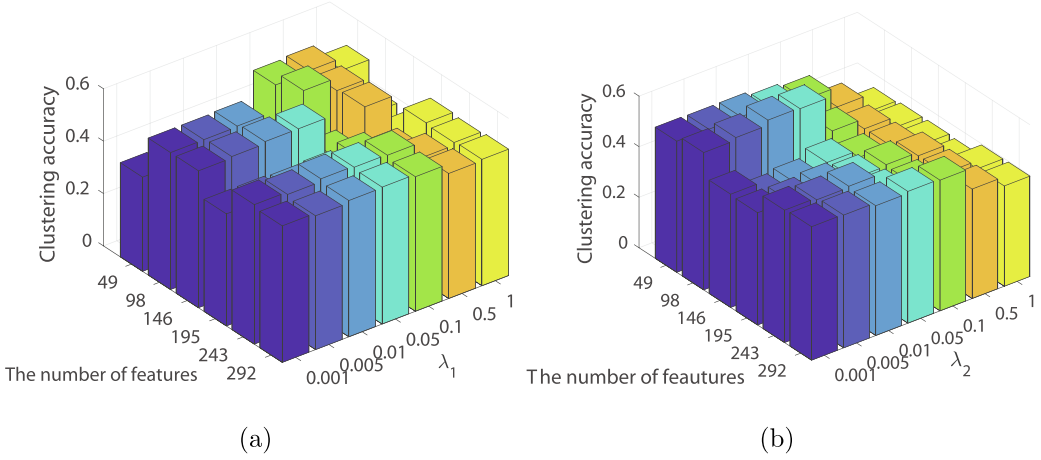

Fig. 2. The clustering accuracy of our method varies with and parameters and the number of selected features. (a) λ_1. (b) λ_2.

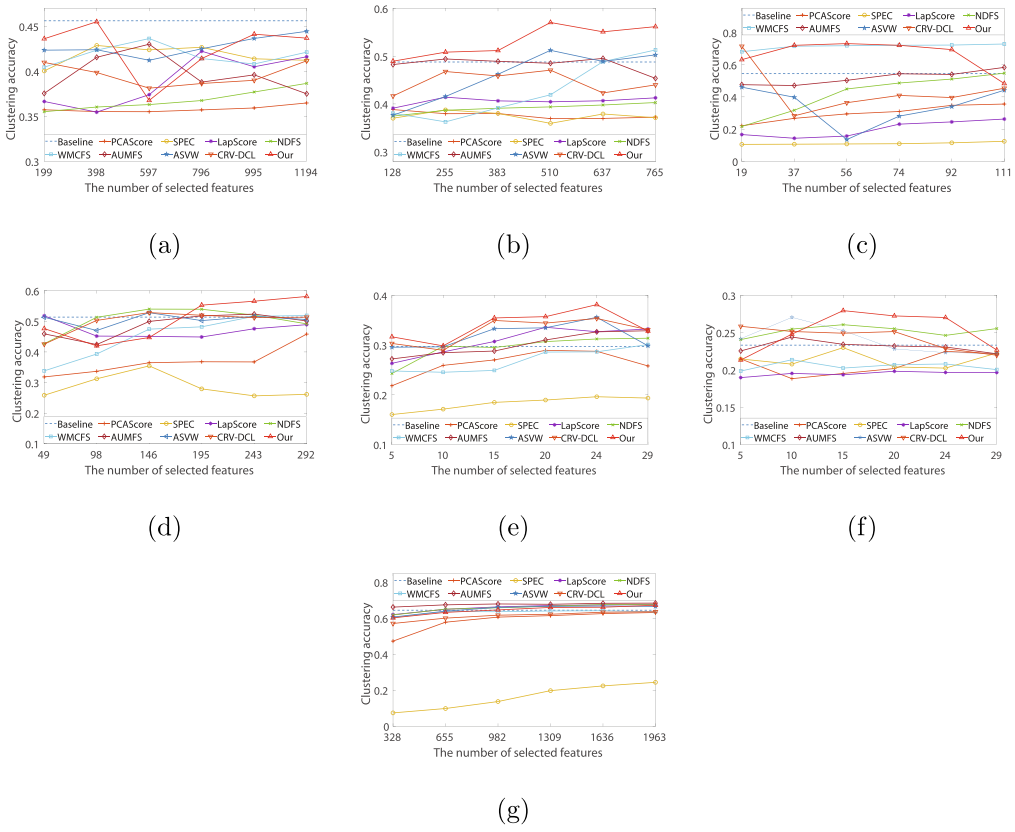

Fig. 3. The ACC results by performing K-means on seven data-sets with different number of selected features. (a) 3Sources. (b) BBCSport. (c) Digital. (d) MSRC-v1. (e) Scene-15. (f) Texture25. (g) Animal.

Table 2
The best ACC results of ten approaches on seven databases. The first row in each cell reports the optimal dimension (·) and the second row in each cell reports the average accuracy over all dimensions.

Methods	Databases						
	3Sources	BBCSport	Digitour	MSRC-v1	Scene-15	Texture25	ANIMAL
Baseline	**45.62 ± 7.77**	48.78 ± 10.49	54.46 ± 2.77	51.39 ± 2.73	29.74 ± 0.70	23.32 ± 0.86	64.41 ± 1.32
PCAScore	36.49 ± 2.81 (1194) 35.82	38.82 ± 2.14 (128) 37.66	35.57 ± 1.26 (111) 29.92	45.84 ± 3.42 (292) 36.90	28.95 ± 1.10 (20) 26.38	22.50 ± 0.78 (24) 20.78	63.15 ± 1.81 (1963) 58.78
SPEC	42.88 ± 3.90 (398) 41.77	38.75 ± 3.52 (255) 37.46	12.42 ± 0.30 (111) 11.13	35.45 ± 4.23 (146) 28.71	19.61 ± 0.81 (24) 18.26	22.99 ± 1.17 (15) 21.36	24.34 ± 1.15 (1963) 16.21
LapScor	42.24 ± 5.15 (796) 38.98	41.42 ± 1.68 (255) 40.60	26.26 ± 0.72 (111) 20.07	51.82 ± 1.41 (49) 47.25	33.58 ± 0.67 (20) 30.86	19.83 ± 0.57 (20) 19.51	67.54 ± 1.29 (1963) 65.85
NDFS	38.66 ± 1.53 (1194) 36.83	40.29 ± 2.02 (765) 39.12	54.75 ± 3.22 (111) 42.13	54.01 ± 2.39 (146) 50.54	31.38 ± 0.27 (29) 29.48	26.07 ± 1.19 (15) 25.22	67.75 ± 1.30 (1963) 65.76
WMCFS	43.64 ± 3.79 (597) 41.80	51.30 ± 11.97 (765) 42.59	72.86 ± 1.06 (111) 71.42	51.92 ± 2.04 (292) 45.40	30.10 ± 0.49 (29) 26.93	21.37 ± 0.81 (10) 20.50	64.15 ± 1.31 (1636) 63.21
AUMFS	43.02 ± 5.56 (597) 39.67	49.53 ± 6.30 (637) 48.33	58.35 ± 1.95 (111) 51.95	52.38 ± 1.15 (243) 48.77	32.88 ± 0.97 (29) 30.18	24.41 ± 0.69 (10) 23.14	**68.37 ± 1.57** (1963) 67.64
ASVW	44.45 ± 6.30 (1194) 42.77	51.22 ± 9.42 (510) 45.97	46.05 ± 0.78 (19) 34.26	52.85 ± 2.28 (146) 50.61	35.59 ± 1.00 (24) 31.91	27.05 ± 1.16 (10) 23.98	66.72 ± 1.54 (1636) 65.06
CRV-DCL	41.17 ± 4.00 (1194) 39.64	47.07 ± 6.12 (510) 44.61	71.45 ± 0.59 (19) 43.67	52.87 ± 1.29 (146) 50.06	35.35 ± 0.66 (24) 32.85	25.85 ± 1.11 (5) 24.32	63.56 ± 1.62 (1963) 61.30
Our	**45.51 ± 5.34** (398) 42.52	**57.05 ± 9.91** (510) 53.22	**73.11 ± 2.57** (56) 66.35	**58.14 ± 2.57** (292) 50.73	**38.13 ± 0.54** (24) 33.94	**27.95 ± 0.86** (15) 25.17	67.19 ± 2.14 (1963) 64.54

Table 3
The best NMI results of ten approaches on seven databases. The first row in each cell reports the optimal dimension (·) and the second row in each cell reports the average accuracy over all dimensions.

	Databases						
Methods	3Sources	BBCSport	Digitour	MSRC-v1	Scene-15	Texture25	ANIMAL
Baseline	24.09 ± 8.51	21.70 ± 14.77	50.80 ± 1.62	41.11 ± 0.66	28.77 ± 0.26	34.68 ± 0.53	73.40 ± 0.46
PCAScore	7.54 ± 2.93 (1194) 7.05	5.44 ± 3.24 (128) 4.43	30.11 ± 1.11 (92) 24.35	36.45 ± 1.65 (292) 27.01	25.03 ± 0.77 (20) 22.19	33.84 ± 0.57 (29) 31.28	72.74 ± 0.49 (1963) 69.93
SPEC	21.88 ± 4.26 (796) 19.15	9.21 ± 2.29 (383) 7.94	2.81 ± 0.26 (111) 1.57	17.62 ± 4.09 (146) 11.07	14.95 ± 0.36 (29) 11.90	33.68 ± 0.92 (15) 31.25	30.07 ± 0.73 (1963) 18.75
LapScor	17.51 ± 7.20 (1194) 13.09	9.09 ± 2.91 (765) 7.91	21.16 ± 0.55 (111) 14.19	42.45 ± 0.77 (49) 38.84	34.33 ± 0.39 (20) 32.28	30.01 ± 0.46 (15) 29.85	76.37 ± 0.32 (1963) 75.18
NDFS	8.24 ± 1.83 (1194) 5.84	7.96 ± 2.77 (765) 6.56	45.76 ± 2.22 (111) 32.52	45.09 ± 2.63 (195) 42.73	32.27 ± 0.10 (29) 30.54	34.55 ± 0.99 (15) 33.73	76.16 ± 0.31 (1963) 74.73
WMCFS	20.01 ± 4.81 (995) 18.98	29.65 ± 14.46 (765) 20.66	68.76 ± 0.86 (111) 67.05	42.18 ±1.66 (292) 35.95	30.29 ±0.32 (29) 25.02	30.46 ± 0.63 (24) 29.10	73.98 ± 0.38 (1636) 73.37
AUMFS	21.42 ± 3.65 (398) 17.87	19.50 ± 6.38 (383) 18.43	51.25 ± 1.36 (111) 44.70	45.74 ± 1.42 (243) 41.41	33.73 ± 0.40 (29) 31.30	32.97 ± 0.78 (24) 32.32	**76.55 ± 0.35** (1963) 75.88
ASVW	20.12 ± 6.44 (1194) 15.60	27.71 ± 12.11 (510) 18.91	39.88 ± 0.41 (19) 25.72	45.55 ± 1.61 (49) 44.20	**35.94 ± 0.59** (24) 34.01	38.03 ± 0.74 (15) 35.71	75.71 ± 0.47 (1963) 74.23
CRV-DCL	18.84 ± 2.05 (199) 18.15	16.94 ± 6.76 (510) 13.30	66.82 ± 0.82 (19) 37.08	42.03 ± 1.18 (146) 39.92	34.57 ± 0.36 (20) 32.58	36.54 ± 0.85 (5) 34.60	74.15 ± 0.49 (1963) 72.41
Our	**26.74 ± 7.09** (1194) 23.19	**39.29 ± 10.37** (510) 36.16	**70.79 ± 1.35** (56) 63.06	**47.99 ± 2.20** (292) 42.40	**35.86 ± 0.61** (20) 34.25	**39.55 ± 0.76** (15) 36.20	75.51 ± 0.54 (1963) 73.87

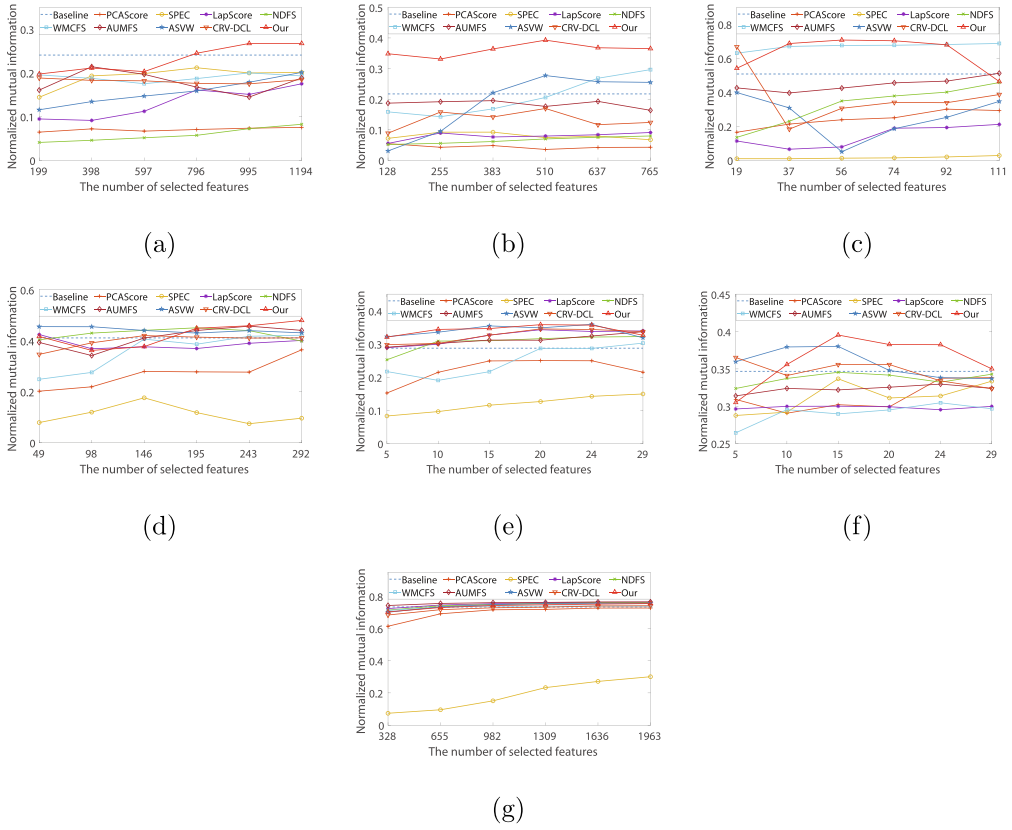

Fig. 4. The NMI results by performing K-means on seven data-sets with different number of selected features. (a) 3Sources. (b) BBCSport. (c) Digital. (d) MSRC-v1. (e) Scene-15. (f) Texture25. (g) Animal.

Table 4
The running time (seconds) per iteration of different multi-view methods.

Methods	Databases						
	3Sources	BBCSport	Digitour	MSRC-v1	Scene-15	Texture25	ANIMAL
WMCFS	0.4474 ± 0.0611	0.4784 ± 0.0290	0.0720 ± 0.0130	0.0550 ± 0.0108	0.0354 ± 0.0044	0.0139 ± 0.0031	7.1531 ± 0.1004
AUMFS	2.0006 ± 0.0398	1.6900 ± 0.0502	0.3780 ± 0.0301	0.1797 ± 0.0107	8.4214 ± 0.6339	0.2081 ± 0.0128	20.8933 ± 0.4102
ASVW	140.3777 ± 10.4725	80.0424 ± 3.2694	1.5579 ± 0.0712	3.4836 ± 0.3653	21.8050 ± 0.1116	0.9763 ± 0.0382	127.9221 ± 7.6980
CRV-DCL	2.8284 ± 0.0638	2.1172 ± 0.1026	0.2958 ± 0.0305	0.2347 ± 0.0339	2.6804 ± 0.2073	0.1522 ± 0.0138	7.9415 ± 0.1195
Our	13.6357 ± 0.4570	8.0706 ± 0.0948	0.7190 ± 0.0363	0.4530 ± 0.0358	20.1975 ± 1.4478	0.6157 ± 0.0665	45.8445 ± 1.3105

method outperform baseline in most cases. (3) Our proposed method can achieve the highest clustering accuracy compared with these feature selection methods in most cases. (3) Through observing the dashed line and comparing the results of the average dimension, we find that our method can obtain the best performance in most cases. It states that our method can choose reasonable features to yield robust experimental results.

4.3. Algorithm performance

The running time of these multi-view methods is reported in Table 4. The experimental results show that WMCFS, AUMFS and CRV-DCL spend less time than ASVW and our proposed method. This may be the need for respectively learning the graph for each view. We further analyze the efficiency among these multi-view methods with a scatter referring to computational time and clustering accuracy. As shown in Fig. 5, although our method takes a little more time for computation, it can get better performance.

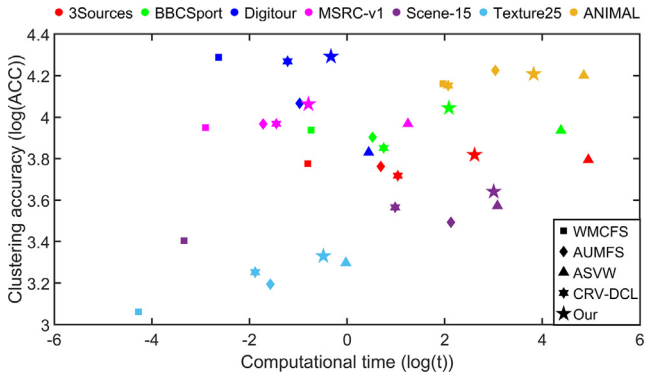

Fig. 5. A scatter plot referring to the computational time and clustering accuracy. Different marks represent different methods while different colours refer to different databases.

In Fig. 6, we show the convergence behaviors of our method and report the convergence curves that record the match error and objective function value in each iteration step. From Fig. 6, one can see that the convergence curves of our proposed methods can well converge, which confirms the effectiveness of our proposed optimization algorithm.

In Fig. 7, we present the illustration of similarity matrices for all the views. In Digitour, we can see that the trends of multi-view graphs are similar to each other. We infer that the consistency is remarkable in Digitour but diversity is still preserved somewhat. On the contrary, the trends of graphs in Texture25 are diverse, which states that the complementary information is important for Texture25. Hence, our proposed method can effectively utilize the consistency and diversity information of multi-view data for feature selection.

5. Conclusion

This paper proposes a tensor low-rank constrained graph embedding model for multi-view unsupervised feature selection, named TLR-MUFS. By adaptively learning the graphs under tensor low-rank constraints, our proposed method can solve the problem of keeping the view-specific information and consistency information in the multi-view graphs simultaneously. The view-specific information is retained in the graph corresponding to each view, while the consistency information is captured by the tensor low-rank graph that consists of all the graphs from different views. To effectively solve our model, an augmented Lagrange multipliers-based optimization algorithm is proposed. Experimental results also demonstrate the effectiveness and superiority of our model for multi-view unsupervised feature selection. In future work, we plan to accelerate graph learning to make it easier to

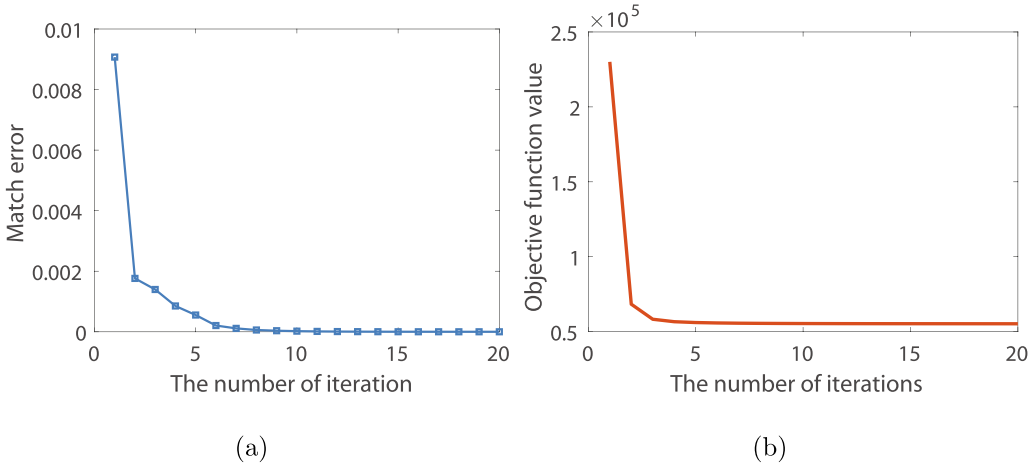

Fig. 6. Convergence curves of our method on MSRC-v1. (a) $Match error = \frac{1}{V}\|\mathbf{S}^{(v)} - \mathbf{G}^{(v)}\|_{\infty}$. (b) Objective function values.

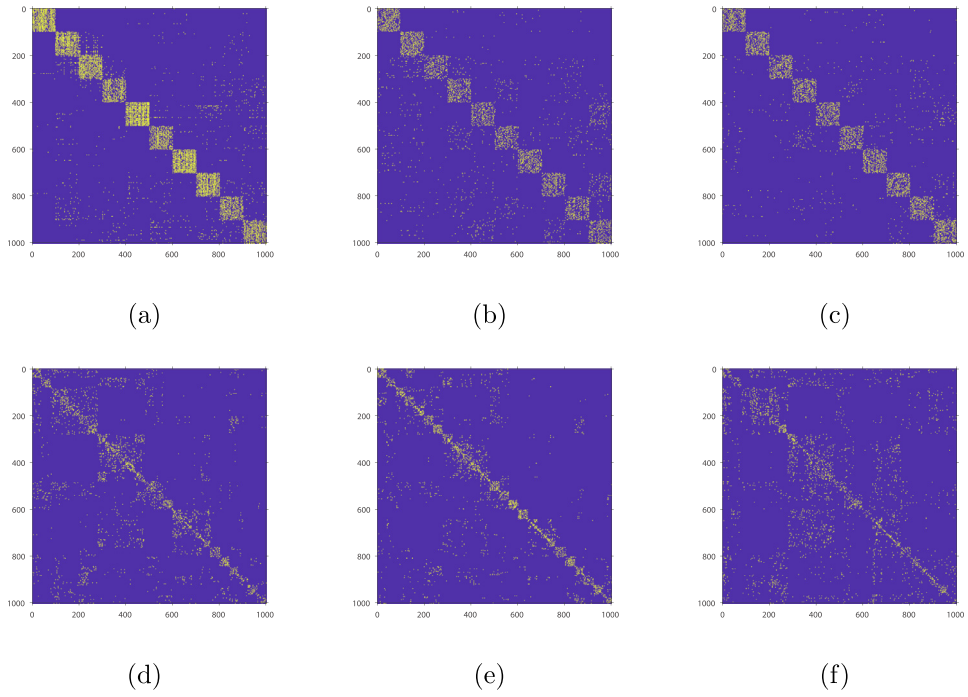

Fig. 7. The illustration of graph matrix $\mathbf{S}^{(v)}$, $v = 1, 2, 3$. (a)-(c) are three views of Digitour. (d)-(f) are three views of Texture25.

apply to large-scale databases. Besides, the number of clusters is a strong prior for unsupervised learning. An unsupervised model with the capability of learning the number of clusters should be considered in the near future.

CRediT authorship contribution statement

Haoliang Yuan: Conceptualization, Methodology, Formal analysis, Writing - original draft. **Junyu Li:** Data curation, Software, Validation, Writing - review & editing. **Yong Liang:** Supervision, Funding acquisition, Writing - review & editing. **Yuan Yan Tang:** Supervision, Funding acquisition, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This research work was supported in part by a grant from the "Macao Young Scholars Program" (Project code: AM201915), in part by the National Nature Science Foundation of China under Grant 61903091 and 62172458, in part by Guangdong Basic and Applied Basic Research Foundation (No. 2020A1515010801), and in part by the Science and Technology Development Fund, Macau SAR (No. 0056/2020/AFJ, 0158/2019/A3).

References

[1] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In Advances in Neural Information Processing Systems, pages 585–591, 2002.
[2] Christopher Bishop. Neural Networks for Pattern Recognition. Oxford University Press, January 1995.
[3] Stephen Boyd, Lieven Vandenberghe, Convex optimization, 03, Cambridge University Press, 2004.
[4] Jian Feng Cai, Emmanuel J. Candès, Zuowei Shen, A singular value thresholding algorithm for matrix completion, SIAM J. Optim. 20 (4) (2010) 1956–1982.
[5] Bokai Cao, Lifang He, Xiangnan Kong, S. Yu Philip, Zhifeng Hao, Ann B. Ragin, Tensor-based multi-view feature selection with applications to brain diseases, in: Proceedings of the IEEE International Conference on Data Mining, IEEE, 2014, pp. 40–49.
[6] Mansheng Chen, Ling Huang, Changdong Wang, Dong Huang, Multi-view clustering in latent embedding space, Proceedings of the AAAI conference on artificial intelligence 34 (2020) 3513–3520.
[7] Xiaojun Chen, Joshua Zhexue Haung, Feiping Nie, Renjie Chen, and Qingyao Wu. A self-balanced min-cut algorithm for image clustering. In Proceedings of the IEEE International Conference on Computer Vision, pages 2061–2069, 2017.
[8] Yiuming Cheung, Hong Zeng, Local kernel regression score for selecting features of high-dimensional data, IEEE Trans. Knowl. Data Eng. 21 (12) (2009) 1798–1802.
[9] Dengxin Dai, Luc Van Gool, Ensemble projection for semi-supervised image classification, in: In Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 2072–2079.
[10] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pages 886–893. IEEE, 2005.
[11] Yinfu Feng, Jun Xiao, Yueting Zhuang, and Xiaoming Liu. Adaptive unsupervised multi-view feature selection for visual concept recognition. In Proceedings of the Asian Conference on Computer Bision, pages 343–357. Springer, 2012.
[12] Derek Greene, Pádraig Cunningham, Practical solutions to the problem of diagonal dominance in kernel document clustering, in: Proceedings of the 23rd International Conference on Machine Learning, ACM Press, 2006, pp. 377–384.
[13] Isabelle Guyon, André Elisseeff, An introduction to variable and feature selection, J. Mach. Learn. Res. (2003) 1157–1182.
[14] Xiaofei He, Deng Cai, and Partha Niyogi. Laplacian score for feature selection. In Advances in Neural Information Processing Systems, pages 507–514, 2006.
[15] Chenping Hou, Feiping Nie, Hong Tao, Dongyun Yi, Multi-view unsupervised feature selection with adaptive similarity and view weight, IEEE Trans. Knowl. Data Eng. 29 (9) (2017) 1998–2011.
[16] Ling Huang, Hong-Yang Chao, Chang-Dong Wang, Multi-view intact space clustering, Pattern Recogn. 86 (2019) 344–353.
[17] Shudong Huang, Zhao Kang, Ivor W Tsang, Xu. Zenglin, Auto-weighted multi-view clustering via kernelized graph learning, Pattern Recogn. 88 (2019) 174–184.
[18] Shudong Huang, Ivor Tsang, Zenglin Xu, Jian Cheng Lv, Measuring diversity in graph learning: A unified framework for structured multi-view clustering, IEEE Trans. Knowl. Data Eng. (2021), 1–1.
[19] Hong Jia, Yiuming Cheung, Subspace clustering of categorical and numerical data with an unknown number of clusters, IEEE Trans. Neural Networks Learn. Syst. 29 (8) (2017) 3308–3325.

[20] Meina Kan, Shiguang Shan, Haihong Zhang, Shihong Lao, Xilin Chen, Multi-view discriminant analysis, IEEE Trans. Pattern Anal. Mach. Intell. 38 (1) (2016) 188–194.
[21] Zhao Kang, Zipeng Guo, Shudong Huang, Siying Wang, Wenyu Chen, Yuanzhang Su, and Zenglin Xu. Multiple partitions aligned clustering. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, pages 2701–2707, 7 2019.
[22] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 2, pages 2169–2178. IEEE, 2006.
[23] Ruihuang Li, Changqing Zhang, Fu. Huazhu, Xi Peng, Tianyi Zhou, Qinghua Hu, Reciprocal multi-layer subspace learning for multi-view clustering, in: In Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 8172–8180.
[24] Zechao Li, Yi Yang, Jing Liu, Xiaofang Zhou, Hanqing Lu, Unsupervised feature selection using nonnegative spectral analysis, in: In Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, 2012.
[25] Zhouchen Lin, Minming Chen, and Yi Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. Computing Research Repository, abs/1009.5055, 2010.
[26] Jialu Liu, Chi Wang, Jing Gao, Jiawei Han, Multi-view clustering via joint nonnegative matrix factorization, in: In Proceedings of the SIAM International Conference on Data Mining, SIAM, 2013, pp. 252–260.
[27] Feiping Nie, Guohao Cai, Jing Li, Xuelong Li, Auto-weighted multi-view learning for image clustering and semi-supervised classification, IEEE Trans. Image Process. 27 (3) (2017) 1501–1511.
[28] Feiping Nie, Heng Huang, Xiao Cai, and Chris Ding. Efficient and robust feature selection via joint ℓ2,1-norms minimization. In Advances in Neural Information Processing Systems, pages 1813–1821, 2010.
[29] Aude Oliva, Antonio Torralba, Modeling the shape of the scene: A holistic representation of the spatial envelope, Int. J. Comput. Vision 42 (3) (2001) 145–175.
[30] Xi Peng, Zhenyu Huang, Jiancheng Lv, Hongyuan Zhu, and Joey Tianyi Zhou. Comic: Multi-view clustering without parameter selection. In Proceedings of the International Conference on Machine Learning, pages 5092–5101. PMLR, 2019.
[31] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary R Bradski. Orb: An efficient alternative to sift or surf. In Proceedings of the International Conference on Computer Vision, volume 11, page 2, 2011.
[32] Wenbin Shao, Abdesselam Bouzerdoum, Son Lam Phung, Signal classification for ground penetrating radar using sparse kernel feature selection, IEEE J. Selected Top. Appl. Earth Observ. Remote Sens. 7 (12) (2014) 4670–4680.
[33] Chang Tang, Xiao Zheng, Xinwang Liu, Wei Zhang, Jing Zhang, Jian Xiong, and Lizhe Wang. Cross-view locality preserved diversity and consensus learning for multi-view unsupervised feature selection. IEEE Trans. Knowl. Data Eng., pp. 1–1, 2021.
[34] Jiliang Tang, Salem Alelyani, and Huan Liu. Feature selection for classification: A review, pages 37–64. CRC Press, 2014.
[35] Yang Wang, Wu Lin, Beyond low-rank representations: Orthogonal clustering basis reconstruction with optimized graph structure for multi-view spectral clustering, Neural Networks 103 (2018) 1–8.
[36] Zhao Wang, Yinfu Feng, Tian Qi, Xiaosong Yang, and Jian J Zhang. Adaptive multi-view feature selection for human motion retrieval. Signal Processing, 120:691–701, 2016.
[37] John Winn and Nebojsa Jojic. Locus: learning object classes with unsupervised segmentation. In Proceedings of the Tenth IEEE International Conference on Computer Vision, vol. 1, pages 756–763. IEEE, 2005.
[38] Deyan Xie, Quanxue Gao, Qianqian Wang, Xiangdong Zhang, Xinbo Gao, Adaptive latent similarity learning for multi-view clustering, Neural Networks 121 (2020) 409–418.
[39] Yuan Xie, Dacheng Tao, Wensheng Zhang, Yan Liu, Lei Zhang, Qu. Yanyun, On unifying multi-view self-representations for clustering by tensor multi-rank minimization, International Journal of Computer Vision 126 (11) (2018) 1157–1179.
[40] Chang Xu, Dacheng Tao, and Chao Xu. A survey on multi-view learning. Computing Research Repository, abs/1304.5634, 2013.
[41] Xinxing Xu, Wen Li, Dong Xu, and Ivor W Tsang. Co-labeling for multi-view weakly labeled learning. IEEE Trans. Pattern Anal. Mach. Intell., 38(6):1113–1125, 2016.
[42] Xu Yumeng, Changdong Wang, Jianhuang Lai, Weighted multi-view clustering with feature selection, Pattern Recogn. 53 (2016) 25–35.
[43] Chao Yao, Yafeng Liu, Bo Jiang, Jungong Han, Junwei Han, Lle score: A new filter-based unsupervised feature selection method based on nonlinear manifold embedding and its application to image recognition, IEEE Trans. Image Process. 26 (11) (2017) 5257–5269.
[44] Shuangyan Yi, Yingyi Liang, Zhenyu He, Yi Li, Wei Liu, Yiu-ming Cheung, Dual pursuit for subspace learning, IEEE Trans. Multimedia 21 (6) (2019) 1399–1411.
[45] Hong Zeng, Yiu-Ming Cheung, A new feature selection method for gaussian mixture clustering, Pattern Recogn. 42 (2) (2009) 243–250.
[46] Hong Zeng, Yiu-ming Cheung, Feature selection and kernel learning for local learning-based clustering, IEEE Trans. Pattern Anal. Mach. Intell. 33 (8) (2010) 1532–1547.
[47] Jing Zhao, Xijiong Xie, Xu Xin, Shiliang Sun, Multi-view learning overview: Recent progress and new challenges, Inform. Fusion 38 (2017) 43–54.
[48] Zheng Zhao, Huan Liu, Spectral feature selection for supervised and unsupervised learning, in: In Proceedings of the 24th International Conference on Machine Learning, ACM, 2007, pp. 1151–1157.

Haoliang Yuan received the B.Sc. and M.Sc. degrees from the Hubei University, Wuhan, China, in 2009 and 2012, and the Ph.D. degree from the University of Macau, 2016. Currently, he is working at Macau University of Science and Technology.

Junyu Li received the B.Eng. and M.Eng. degrees from the Guangdong University of Technology, Guangzhou, China, in 2017 and 2020. Currently, he is pursuing the Ph.D. degree in South China University of Technology.

Yong Liang received the B.S. and M.S. degrees in Applied Mathematics from Xi'an Jiaotong University, China, in 1996 and 1999, and the Ph.D. degree in Computer Science from the Chinese University of Hong Kong in 2003. He was with the Chinese University of Hong Kong (2004–2007) as a post doctor fellow. He is an Assistant Professor, Associate Professor and Professor at Macau University of Science and Technology. His research interests include machine learning, data mining, and bioinformatics.

Yuan Yan Tang received the B.S. degree in electrical and computer engineering from Chongqing University, Chongqing, China, the M.Eng. degree in electrical engineering from the Beijing Institute of Post and Telecommunications, Beijing, China, and the Ph.D. degree in computer science from Concordia University, Montreal, QC, Canada. He is currently a Chair Professor with the Faculty of Science and Technology, University of Macau, Macau, China, and a Professor/Adjunct Professor/Honorary Professor with several institutes, including several universities in China, Concordia University, Canada, and Hong Kong Baptist University, Hong Kong. He has published over 400 technical papers and authored/co-authored over 25 monographs/books/bookchapters on subjects ranging from electrical engineering to computer science. His current research interests include wavelet theory and applications, pattern recognition, image processing, document processing, artificial intelligence, and Chinese computing. Dr. Tang is the Founder and the Editor-in-Chief of the International Journal on Wavelets, Multiresolution, and Information Processing, and an Associate Editor of several international journals, such as the International Journal on Pattern Recognition and Artificial Intelligence. He is the Founder and the Chair of Pattern Recognition Committee in the IEEE SYSTEMS, MAN, AND CYBERNETICS. He has served as the General Chair, the Program Chair, and a Committee Member for many international conferences, including the General Chair of the 18th International Conference on Pattern Recognition. He is the Founder and the General Chair of the series International Conferences on Wavelets Analysis and Pattern Recognition. He is a fellow of the International Associate of Pattern Recognition. 10