Check for updates

Consensus Affinity Graph Learning via Structure Graph Fusion and Block Diagonal Representation for Multiview Clustering

Zhongyan Gui[1,2] · Jing Yang[1] · Zhiqiang Xie[3] · Cuicui Ye[1]

Abstract

Learning a robust affinity graph is fundamental to graph-based clustering methods. However, some existing affinity graph learning methods have encountered the following problems. First, the constructed affinity graphs cannot capture the intrinsic structure of data well. Second, when fusing all view-specific affinity graphs, most of them obtain a fusion graph by simply taking the average of multiple views, or directly learning a common graph from multiple views, without considering the discriminative property among diverse views. Third, the fusion graph does not maintain an explicit cluster structure. To alleviate these problems, the adaptive neighbor graph learning approach and the data self-expression approach are first integrated into a structure graph fusion framework to obtain a view-specific structure affinity graph to capture the local and global structures of data. Then, all the structural affinity graphs are weighted dynamically into a consensus affinity graph, which not only effectively incorporates the complementary affinity structure of important views but also has the capability of preserving the consensus affinity structure that is shared by all views. Finally, a k–block diagonal regularizer is introduced for the consensus affinity graph to encourage it to have an explicit cluster structure. An efficient optimization algorithm is developed to tackle the resultant optimization problem. Extensive experiments on benchmark datasets validate the superiority of the proposed method.

Keywords Adaptive neighbor graph · Self-expression · Structure graph fusion · Consensus affinity graph · K-block diagonal regularizer

✉ Jing Yang
  yangjing@hrbeu.edu.cn

1 College of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China

2 College of Computer Science and Technology, Heilongjiang University, Harbin 150080, China

3 College of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China

🙾 Springer

1 Introduction

Recently, multiview data have become very common in many real-world applications [1, 2]. For example, in image data processing, an image can be presented by diverse features, such as HOG, SIFT, and LBP [3]. In biometrics, a person's identity can be recognized by faces, fingerprints, and sounds [4]. Since diverse feature views can depict diverse perspectives of the same object, sufficient research results have demonstrated that the performance of multiview learning will be substantially improved by excavating complementary information of multiview data [5].

Multiview clustering (MVC), which divides multiview data into different groups by efficiently integrating multiple feature views to guarantee that highly similar instances are divided into the same group while dissimilar instances are divided into different groups, is a fundamental task of multiview learning [6, 7]. In general, most existing MVC methods employ graph-based models since the similarity graph can effectively characterize the data structure [8]. Typically, these methods first construct a view-specific affinity graph by using some similarity metrics and then fuse all the constructed view-specific affinity graphs into a consensus affinity graph. Finally, a spectral or graph algorithm is applied to realize clustering [9, 10], or the clustering result can be obtained directly from the fusion [11].

The clustering performance of graph-based methods is largely dependent on the quality of the constructed affinity graph [12]. To learn a better affinity graph, various affinity graph construction methods are proposed, and typically, they are divided into three categories [13]. The first category is to predefine a similarity graph as an affinity graph [14]. Here, some outstanding issues are as follows: (1) The construction of an affinity graph is easily affected by the choice of similarity metric, neighborhood size, and scaling parameter, all of which are data-dependent and noise-sensitive [15]; (2) The constructed affinity graph cannot capture the underlying graph structure of the data well. The second category is the adaptive neighbors graph approach, which assigns adaptive and optimal neighbors to each data point according to local distances to learn an affinity graph [16–18]. It does not need to specify the neighborhood size, and the similarity among data points is adaptively learned from the data. Generally, the data points with a smaller distance have a higher affinity value, while the data points with a larger distance have a lower affinity value. This approach is an effective way to preserve the local manifold structure [15]. The third category is the data self-expression approach, which reconstructs each data point by a linear combination of all other points in the same subspace and then generates a coefficient matrix to build the affinity graph [19, 20]. This approach is an effective way to capture the global structure [21].

In general, the underlying data structures are unknown in advance which poses a challenge for constructing an affinity graph that can best capture the essential data structure [22]. Because both the local and global graph structures are crucial for uncovering the graph structures of data, they can provide each other with possible complementary information to boost the performance. Therefore, it is necessary to integrate the adaptive neighbor graph approach and the data self-expression approach into a unified graph structure learning model to learn a view-specific affinity graph that can not only automatically learn the similarity information from the data but also capture the local and global structures of the data.

Given that diverse views admit the same underlying cluster structure, we can obtain the consensus information from the fusion of diverse views to better exploit the cluster structure [23]. Therefore, after obtaining the view-specific affinity graphs, we need to consider how to effectively fuse them. Simply taking the average of them [24] or directly generating a common graph from them [25] fails to consider the discrimination of different views. Kang

et al. [26] proposed a new dynamically weighted graph fusion method to integrate multiview information, and each view can be treated as perturbations of a consensus graph. Usually, the closer it is to the consensus graph, the larger the weight that is assigned. This graph fusion method can merge different views into a consensus graph, distinguish the contributions of different views and explore the heterogeneous complementary information effectively.

An ideal consensus affinity graph should have between-cluster affinities that are all zeros, while the inner-cluster affinities are not zeros, namely, it should obey the block diagonal structure, which facilitates good clustering performance [27–30]. Unfortunately, in real-life noisy applications, the consensus affinity graph may only have a weak block-diagonal structure, and the number of target blocks in it is difficult to control. Thus, we need to consider how to build a robust block diagonal representation for the consensus affinity graph. Because the k-block diagonal regularizer is a method to pursue a block diagonal structure directly containing k blocks [27], it encourages the affinity graph to obey the desired k-block diagonal structure, and therefore, it is essential to employ the k-block diagonal regularizer to improve the quality of the consensus affinity graph.

In summary, a novel MVC method, namely consensus affinity graph learning via structure graph fusion and block diagonal representation (CAGL-SGBD) for multiview clustering, has been proposed, and the main contributions of this work are as follows:

(1) Adaptive neighbor graph learning and the data self-expression model are integrated into a unified structure graph fusion framework that can capture the local and global structures of the data, be robust to noise, and guide the construction of the initial affinity graph for each individual view.

(2) All the constructed structure affinity graphs are weighted into a consensus affinity graph that not only incorporates the complementary affinity structure of important views but also encourages the learned consensus affinity graph to have the capability of preserving the consensus affinity structure that is unanimously admitted by multiple views.

(3) The k-block diagonal regularizer is introduced for the consensus affinity graph to force it to have an explicit cluster structure.

(4) Our method integrates structure graph fusion, consensus affinity graph learning, and k-block diagonal regularization, which helps to obtain an enhanced consensus affinity graph that maintains the graph structure characteristics of multiview data and is beneficial to clustering.

The rest of the paper is organized as follows. Section 2 briefly introduces the preliminaries that are necessary for the research. Section 3 gives a detailed description and formulation of CAGL-SGBD. Section 4 designs an efficient optimization algorithm, and some analyses are presented. Section 5 presents numerical experiments. Section 6 concludes our work.

2 Preliminaries

In this subsection, we first present notations used throughout the paper, and then briefly overview some technologies that are necessary for this work.

2.1 Notations

In this paper, matrices are denoted as boldface capital letters, e.g., X. Vectors are written as boldface lower-case letters, e.g., x, and scalars are written as lower-case letters, e.g., x. For

an arbitrary matrix X, its (i, j)-th entry is written as x_{ij}, its j-th column is written as x_j, and its i-th row is written as x^i. For x, its i-th entry is written as x_i. The transpose and inverse of matrix X are denoted as X^T and X^{-1} respectively. The data matrix of the v-th view is denoted as $X^{(v)}$. $\|X\|_2^2$ and $\|X\|_F^2$ represent the l_2-norm and Frobenius norm of matrix X, respectively. The identity matrix is denoted by I.

2.2 Local Structure Graph Learning

Recently, the adaptive neighbor graph approach has been widely employed in graph-based clustering to capture the local manifold structure [18, 21, 31]. Given n multiview observations $\{[x_i^{(1)}; \cdots ; x_i^{(n_v)}]\}_{i=1}^n$ from n_v different views, $x_i^{(v)}$ denotes the i-th data point of the v-th view, and $X^{(v)} = [x_1^{(v)}, x_2^{(v)}, \cdots, x_n^{(v)}]$. $Z^{(v)}$ is the affinity matrix of the v-th view. The adaptive neighbors graph approach can be expressed as

$$\min_{Z^{(v)}} \sum_{v=1}^{n_v} \sum_{i,j=1}^n \|x_i^{(v)} - x_j^{(v)}\|_2^2 z_{ij}^{(v)}$$
$$s.t. \ Z^{(v)}\mathbf{1} = \mathbf{1}, \ Z^{(v)} \geq 0, \tag{1}$$

In Eq. (1), the constraint term is added to guarantee that the sum of each row of $Z^{(v)}$ is one and to ensure the probability property of $Z^{(v)}$ [32]. The affinity matrices learned from Eq. (1) will capture the local manifold structure adaptively. Since the local structure is very prominent for its information discovery ability, and generally, it is believed to be better than the global structure [33], therefore, exploring the local structure graph learning is a widely recognized graph clustering method [34, 35]. However, it is susceptible to noise and ignores the global structure of data.

2.3 Global Structure Graph Learning

The data self-expression approach is an effective way to automatically capture the global structure of data [22], and its basic idea is that each data sample can be expressed as a linear combination of other samples, and the combination coefficient indicates the similarities between samples [26]. The multiview self-expression model can be expressed as

$$X^{(v)} = X^{(v)}Z^{(v)} + E^{(v)}, v \in \{1, \cdots, n_v\} \tag{2}$$

where $X^{(v)}$, $Z^{(v)}$, and $E^{(v)}$ stand for the data matrix, coefficient matrix, and error matrix for the v-th view, respectively. The coefficient matrix $Z^{(v)}$ characterizes the similarities among samples, a low-rank constraint can be applied to it to capture the global data structure [36], and a $l_{2,1}$-norm can be imposed on $E^{(v)}$ to address the sample-specific outliers and corruptions [37]. Thus, the global structure graph learning can be formulated as

$$\min_{Z^{(v)}, E^{(v)}} \sum_{v=1}^{n_v} \|Z^{(v)}\|_* + \lambda \sum_{v=1}^{n_v} \|E^{(v)}\|_{2,1}$$
$$s.t. \ X^{(v)} = X^{(v)}Z^{(v)} + E^{(v)}, Z^{(v)} \geq 0 \tag{3}$$

Equation (3) is also a typical objective function of subspace clustering based on the self-representation model, and the coefficient matrix $Z^{(v)}$ learned from it can capture the global structure information of the v-th view. Many existing subspace clustering methods are based

on the above model to cooperatively learn the low-rank representation and the affinity matrix to improve the clustering performance [38, 39].

2.4 *K*-block Diagonal Regularizer

The k-block diagonal structure of the affinity graph can promote perfect data clustering. For an arbitrary affinity matrix $H \in R^{n \times n} (H \geq 0, H = H^T)$, its corresponding Laplacian matrix is $L_H = Diag(H1) - H$, The k-block diagonal regularizer [30, 40] is defined as

$$\| H \|_{\boxed{k}} = \sum_{i=n-k+1}^{n} \lambda_i(L_H) \tag{4}$$

where $\lambda_i(L_H)$ ($i \in \{1, \cdots, n\}$) are the eigenvalues of L_H in descending order.

The regularizer in Eq. (4) has an important property, namely, the multiplicity k of the eigenvalue 0 of L_H equals the number of connected components (blocks) in H, which will make the affinity graph have exactly k connected components for data with k clusters. Thus, the k-block diagonal regularizer can not only facilitate the affinity matrix to be a block diagonal structure but also control the number of blocks [29, 30].

3 Proposed Method

3.1 Structure Graph Fusion

To explore a more flexible local manifold structure and a better global structure representation capacity and to take full advantage of the possible complementary information provided by them, we integrate Eqs. (1) and (3) into a unified structure graph fusion framework to jointly learn a view-specific structure affinity graph, which can be formally expressed as

$$\min_{Z^{(v)}, E^{(v)}} \sum_{v=1}^{n_v} \{Tr(X^{(v)} L_Z^{(v)} X^{(v)T}) + \lambda_1 \|Z^{(v)}\|_* + \lambda_2 \|E^{(v)}\|_{2,1}\}$$
$$s.t. \; X^{(v)} = X^{(v)} Z^{(v)} + E^{(v)}, \; Z^{(v)} 1 = 1, \; Z^{(v)} \geq 0 \tag{5}$$

where λ_1 and λ_2 are trade-off parameters. $L_Z^{(v)}$ denotes the Laplacian matrix, $L_Z^{(v)} = D_Z^{(v)} - (Z^{(v)T} + Z^{(v)})/2, D_Z^{(v)}$ is a diagonal degree matrix, and its diagonal elements are $\sum_j (z_{ij}^{(v)} + z_{ji}^{(v)})/2$. The first term in Eq. (5) ensures that each entry in $Z^{(v)}$ can directly describe the local similarity between data points in the v-th view, and the second term encourages $Z^{(v)}$ to follow the low-rank property to capture the global structure of the data. The third term addresses sample-specific corruption and outliers.

Thus, the structure affinity graph $Z^{(v)}$ learned from Eq. (5) can not only characterize the affinities between data points in the v-th view but also preserve the local and global structures of the data in the v-th view.

3.2 Consensus Affinity Graph Learning

To learn an optimal consensus affinity graph, we fuse all the structure affinity graphs into a consensus affinity graph S based on two intuitive assumptions [26]: (1) Each $Z^{(v)}$ can be

regarded as a perturbation of S; (2) The closer $\mathbf{Z}^{(v)}$ is to S, the larger the weight of the v–th view is assigned. Thus, we have

$$\min_{S} \sum_{v=1}^{n_v} w^{(v)} \|S - \mathbf{Z}^{(v)}\|_F^2$$

$$s.t.\ \mathbf{s}^i \mathbf{1} = 1,\ s_{ij} \geq 0,\ w^{(v)} \geq 0 \tag{6}$$

where $w^{(v)}$ is the weight of the v–th view and represents the importance of the v–th view; the larger $w^{(v)}$ is, the greater the importance of the v–th view. $\mathbf{s}^i$ represents the i–th row of S, and the constraint terms are added to ensure the probabilistic nature of S.

To ensure that learned consensus affinity graph S can well characterize the affinities between data points from different views and capture the consensus affinity structure that is universally admitted by all views, we assume that for all structure affinity graphs $\{\mathbf{Z}^{(v)}\}_{v=1}^{n_v}$, any two data points z_i and z_j should have the same affinity value s_{ij}. Then, we have

$$\min_{S} \frac{1}{2} \sum_{v=1}^{n_v} \sum_{i,j}^{n} \|z_i^{(v)} - z_j^{(v)}\|_2^2 s_{ij}$$

$$s.t.\ s_{ij} \geq 0 \tag{7}$$

We employ Eq. (7) to learn the similarities between data points is based on an intuitive assumption that the self-expressiveness coefficient matrix $\mathbf{Z}^{(v)}$ learned from Eq. (5) can be deemed the substitute of the data matrix $X^{(v)}$ because each entry of $\mathbf{Z}^{(v)}$ can quantify the similarity between two data points in $X^{(v)}$, namely, if two data points are close to each other in the original space, their new representations in the new space must also be similar to each other [13]. Furthermore, compared to the original data matrix $X^{(v)}$, the clean structure affinity matrix $\mathbf{Z}^{(v)}$ can better describe the intrinsic structure of the real data; thus, a more robust consensus affinity matrix S can be derived from $\{\mathbf{Z}^{(v)}\}_{v=1}^{n_v}$.

Then, the model of consensus affinity graph learning is

$$\min_{S, w^{(v)}} \sum_{v=1}^{n_v} \{w^{(v)} \|S - \mathbf{Z}^{(v)}\|_F^2 + Tr(\mathbf{Z}^{(v)} \mathbf{L}_S \mathbf{Z}^{(v)T})\}$$

$$s.t.\ \mathbf{s}^i \mathbf{1} = 1,\ s_{ij} \geq 0,\ w^{(v)} \geq 0 \tag{8}$$

where $\mathbf{L}_S$ denotes the Laplacian matrix, $\mathbf{L}_S = \mathbf{D}_S - (S^T + S)/2$, and $\mathbf{D}_S$ is a diagonal degree matrix whose diagonal elements are $\sum_j (s_{ij} + s_{ji})/2$. Equation (8) can adaptively learn a consensus affinity graph S that has the capability of preserving the consensus affinity structure being admitted by all the structural affinity graphs. At the same time, during the fusion process, the structural affinity graph is dynamically weighted, which effectively reduces the influences of noisy views and merges important complementary information.

3.3 Block Diagonal Representation

However, the consensus affinity graph S learned from Eq. (8) may not have the block diagonal structure that is needed for clustering. Therefore, we introduce a k-block diagonal regularizer

for S to ensure that it satisfies the block diagonal characteristic; then, we have

$$\|S\|_{\boxed{k}} = \sum_{i=n-k+1}^{n} \lambda_i(L_S)$$
$$s.t.\, S \geq 0,\, S = S^T \tag{9}$$

where $S = [s_1, \cdots, s_j, \cdots, s_n]$. The S learned from Eq. (9) is k-block diagonal and has an explicit cluster structure.

3.4 Objective Function

By integrating the structure graph fusion in Eq. (5), the consensus affinity graph learning in Eq. (8), and the block diagonal representation in Eq. (9) into a unified model, the final objective function is

$$\min_{S,w^{(v)},Z^{(v)},E^{(v)}} \sum_{v=1}^{n_v} \{w^{(v)}\|S - Z^{(v)}\|_F^2 + Tr(Z^{(v)}L_S Z^{(v)T}) + 2\lambda_1 Tr(X^{(v)} L_Z^{(v)} X^{(v)T})$$
$$+ \lambda_2\|Z^{(v)}\|_* + \lambda_3\|E^{(v)}\|_{2,1}\} + \lambda_4\|S\|_{\boxed{k}}$$
$$s.t.\, X^{(v)} = X^{(v)}Z^{(v)} + E^{(v)},\, Z^{(v)}\mathbf{1} = \mathbf{1},\, Z^{(v)} \geq 0,$$
$$s^i\mathbf{1} = \mathbf{1},\, S \geq 0,\, S = S^T,\, w^{(v)} \geq 0 \tag{10}$$

where λ_1, λ_2, λ_3 and λ_4 are trade-off parameters for balancing the corresponding terms. The first term fuses the structural affinity graphs $\{Z^{(v)}\}_{v=1}^{n_v}$ of different views adaptively into a consensus affinity graph S. The second term encourages the learned S to capture the affinities among data points that are unanimously admitted by all views. The third term ensures that $Z^{(v)}$ can adaptively capture the local manifold structure of the original data. The fourth term imposes a low-rank constraint on the representation matrix $Z^{(v)}$ to capture the global structure. Through the joint learning of the first four terms, the structural affinity matrix $Z^{(v)}$ can capture both the local and global structures of the data in each individual view, and meanwhile, the intrinsic structures of data that are contained in the structural affinity graphs $Z^{(v)}$ can be well integrated into the consensus affinity graph S. The fifth term resists sample-specific corruptions and outliers to enhance the robustness of the model. The sixth term is the k-block diagonal representation of the learned S.

Consequently, the consensus affinity graph S learned from Eq. (10) can well integrate the underlying data structure of $\{Z^{(v)}\}_{v=1}^{n_v}$, characterize the similarity among data points, and have an explicit k-block diagonal structure.

4 Optimization

In this section, an augmented Lagrange multiplier with an alternating direction minimizing strategy is used to solve Eq. (10). Specifically, Eq. (10) can be optimized alternatively by introducing auxiliary variables: $S = M$, $Z^{(v)} = A^{(v)}$, $Z^{(v)} = B^{(v)}$, $Z^{(v)} = C^{(v)}$ and $Z^{(v)} = D^{(v)}$, $v \in \{1, \ldots, n_v\}$, to make problem (10) separable. Then Eq. (10) is converted to the

following optimization problem:

$$\min_{\substack{S,w^{(v)},E^{(v)},M,Z^{(v)},\\ A^{(v)},B^{(v)},C^{(v)},D^{(v)}}} \sum_{v=1}^{n_v}\{w^{(v)}\|S-A^{(v)}\|_F^2 + Tr(B^{(v)}L_SB^{(v)T})+$$

$$2\lambda_1 Tr(X^{(v)}L_C^{(v)}X^{(v)T}) + \lambda_2\|D^{(v)}\|_* + \lambda_3\|E^{(v)}\|_{2,1}\} + \lambda_4\|M\|_{\boxed{k}}$$

$$s.t.\ X^{(v)} = X^{(v)}Z^{(v)} + E^{(v)}, Z^{(v)} = A^{(v)}, Z^{(v)} = B^{(v)}, Z^{(v)} = C^{(v)},$$

$$Z^{(v)} = D^{(v)}, S = M, C^{(v)}\mathbf{1} = \mathbf{1}, C^{(v)} \geq 0, D^{(v)} \geq 0, M \geq 0,$$

$$M = M^T, s^i\mathbf{1} = 1, s_{ij} \geq 0, w^{(v)} \geq 0 \tag{11}$$

The augmented Lagrange function of Eq. (11) is

$$\min_{\substack{S,w^{(v)},E^{(v)},M,Z^{(v)},\\ A^{(v)},B^{(v)},C^{(v)},D^{(v)}}} \sum_{v=1}^{n_v}\{w^{(v)}\|S-A^{(v)}\|_F^2 + Tr(B^{(v)}L_SB^{(v)T})+$$

$$2\lambda_1 Tr(X^{(v)}L_C^{(v)}X^{(v)T}) + \lambda_2\|D^{(v)}\|_* + \lambda_3\|E^{(v)}\|_{2,1}\} + \lambda_4\|M\|_{\boxed{k}}+$$

$$\sum_{v=1}^{n_v}\frac{\mu}{2}(\|X^{(v)} - X^{(v)}Z^{(v)} - E^{(v)} + \frac{Y_1^{(v)}}{\mu}\|_F^2+$$

$$\|Z^{(v)} - A^{(v)} + \frac{Y_2^{(v)}}{\mu}\|_F^2 + \|Z^{(v)} - B^{(v)} + \frac{Y_3^{(v)}}{\mu}\|_F^2+$$

$$\|Z^{(v)} - C^{(v)} + \frac{Y_4^{(v)}}{\mu}\|_F^2 + \|Z^{(v)} - D^{(v)} + \frac{Y_5^{(v)}}{\mu}\|_F^2) + \frac{\mu}{2}\|S - M + \frac{Y_6}{\mu}\|_F^2$$

$$s.t.C^{(v)}\mathbf{1} = \mathbf{1}, C^{(v)} \geq 0, D^{(v)} \geq 0, M \geq 0, M = M^T, s^i\mathbf{1} = 1, s_{ij} \geq 0, w^{(v)} \geq 0 \tag{12}$$

where $Y_1^{(v)}$, $Y_2^{(v)}$, $Y_3^{(v)}$, $Y_4^{(v)}$,$Y_5^{(v)}$ and Y_6 are Lagrange multipliers, and $\mu > 0$ is a penalty parameter. Equation (12) can be solved by alternately updating each variable while fixing all the other variables. The update rules are as follows:

Update $A^{(v)}$ Fixing all variables except $A^{(v)}$, Eq. (12) can be written as

$$\min_{A^{(v)}} w^{(v)}\|S - A^{(v)}\|_F^2 + \frac{\mu}{2}\|Z^{(v)} - A^{(v)} + \frac{Y_2^{(v)}}{\mu}\|_F^2 \tag{13}$$

Taking the derivative of Eq. (13) w.r.t. $A^{(v)}$ and setting it to 0, the updating rule of $A^{(v)}$ is

$$A^{(v)^{t+1}} = (2w^{(v)} + \mu)^{-1}(2w^{(v)}S + \mu Z^{(v)} + Y_2^{(v)}) \tag{14}$$

Update $B^{(v)}$ Fixing all variables except $B^{(v)}$, Eq. (12) can be simplified as

$$\min_{B^{(v)}} Tr(B^{(v)}L_SB^{(v)T}) + \frac{\mu}{2}\|Z^{(v)} - B^{(v)} + \frac{Y_3^{(v)}}{\mu}\|_F^2 \tag{15}$$

Taking the derivative of Eq. (15) w.r.t. $B^{(v)}$ and setting it to 0, the updating rule of $B^{(v)}$ is

$$B^{(v)^{t+1}} = (\mu Z^{(v)} + Y_3^{(v)})(2L_S + \mu I)^{-1} \tag{16}$$

Update $C^{(v)}$ Fixing all variables except $C^{(v)}$, Eq. (12) can be simplified as

$$\min_{C^{(v)}} \lambda_1 \sum_{i,j=1}^{n} \|x_i^{(v)} - x_j^{(v)}\|_2^2 c_{ij}^{(v)} + \frac{\mu}{2} \|Z^{(v)} - C^{(v)} + \frac{Y_4^{(v)}}{\mu}\|_F^2$$

$$s.t. C^{(v)}\mathbf{1} = \mathbf{1}, C^{(v)} \geq 0 \tag{17}$$

To simplify notations, the view index is tentatively ignored. Let h_{ij} be the (i, j)-th element of $H = (Z + \frac{Y_4}{\mu})$; note that Eq. (17) is independent between different rows, we can address the following problem separately for each i,

$$\min_{c^i \mathbf{1}=1, c_{ij} \geq 0} \lambda_1 \sum_{j=1}^{n} \|x_i - x_j\|_2^2 c_{ij} + \sum_{j=1}^{n} (\frac{\mu}{2} c_{ij}^2 - \mu h_{ij} c_{ij}) \tag{18}$$

Denote $g_{ij} = \lambda_1 \|x_i - x_j\|_2^2 - \mu h_{ij}$ as the j-th element of $g_i \in \mathcal{R}^{1 \times n}$; then, c^i in Eq. (18) can be updated as

$$\min_{c^i \mathbf{1}=1, c^i \geq 0} \|c^i + \frac{g_i}{\mu}\|_2^2 \tag{19}$$

The Lagrangian function of Eq. (19) is

$$\mathcal{L}(c^i, \delta, \varphi) = \|c^i + \frac{g_i}{\mu}\|_2^2 + \delta(1 - c^i \mathbf{1}) + \varphi^T(-c^i) \tag{20}$$

where δ and $\varphi \geq 0$ are the Lagrangian multipliers, and the optimal solution c^i is

$$c^{i^{t+1}} = (-\frac{g_i}{\mu} + \delta \mathbf{1}^T)_+ \tag{21}$$

where $(\cdot)_+ = \max(\cdot, 0)$.

Update $D^{(v)}$ Fixing all variables except $D^{(v)}$, Eq. (12) is equivalent to

$$\min_{D^{(v)}} \frac{\lambda_2}{\mu} \|D^{(v)}\|_* + \frac{1}{2} \|D^{(v)} - (Z^{(v)} + \frac{Y_5^{(v)}}{\mu})\|_F^2 \tag{22}$$

Equation (22) can be solved by the singular value thresholding operator [41].

Update $Z^{(v)}$ Fixing all variables except $Z^{(v)}$, Eq. (12) is equivalent to

$$\min_{Z^{(v)}} \|X^{(v)} - X^{(v)}Z^{(v)} - E^{(v)} + \frac{Y_1^{(v)}}{\mu}\|_F^2 +$$

$$\|Z^{(v)} - A^{(v)} + \frac{Y_2^{(v)}}{\mu}\|_F^2 + \|Z^{(v)} - B^{(v)} + \frac{Y_3^{(v)}}{\mu}\|_F^2 +$$

$$\|Z^{(v)} - C^{(v)} + \frac{Y_4^{(v)}}{\mu}\|_F^2 + \|Z^{(v)} - D^{(v)} + \frac{Y_5^{(v)}}{\mu}\|_F^2 \tag{23}$$

Taking the derivative of Eq. (23) w.r.t. $Z^{(v)}$ and setting it to 0, the updating rule of $Z^{(v)}$ is

$$Z^{(v)^{t+1}} = (X^{(v)T}X^{(v)} + 4I)^{-1}(X^{(v)T}V_1 + V_2 + V_3) \tag{24}$$

where $V_1 = X^{(v)} - E^{(v)} + Y_1^{(v)}/\mu$, $V_2 = A^{(v)} + B^{(v)} + C^{(v)} + D^{(v)}$, $V_3 = (Y_2^{(v)} + Y_3^{(v)} + Y_4^{(v)} + Y_5^{(v)})/\mu$.

Update $E^{(v)}$ Fixing all variables except $E^{(v)}$, Eq. (12) is equivalent to

$$\min_{E^{(v)}} \frac{\lambda_3}{\mu} \|E^{(v)}\|_{2,1} + \frac{1}{2} \|E^{(v)} - F^{(v)}\|_F^2 \tag{25}$$

where $F^{(v)} = X^{(v)} - X^{(v)}Z^{(v)} + Y_1^{(v)}/\mu$. According to [37], if the optimal solution to Eq. (25) is $E^{(v)}$, then the j-th column of $E^{(v)}$ is

$$[E^{(v)^{t+1}}]_{:,j} = \begin{cases} \frac{\|[F^{(v)}]_{:,j}\|_2 - \lambda_3/\mu}{\|[F^{(v)}]_{:,j}\|_2}[F^{(v)}]_{:,j}, & if \|[F^{(v)}]_{:,j}\|_2 > \frac{\lambda_3}{\mu} \\ 0, & otherwise \end{cases} \tag{26}$$

Update S Fixing all variables except S, Eq. (12) is equivalent to

$$\min_S \sum_{v=1}^{n_v} w^{(v)} \|S - A^{(v)}\|_2^2 + \frac{1}{2} \sum_{v=1}^{n_v} \sum_{i,j=1}^n \|b_i^{(v)} - b_j^{(v)}\|_2^2 s_{ij} + \frac{\mu}{2} \|S - M + \frac{Y_6}{\mu}\|_F^2$$

$$s.t. s^i \mathbf{1} = 1, S \geq 0 \tag{27}$$

Equation (27) is further written as

$$\min_{s^i \mathbf{1}=1, s_{ij} \geq 0} \sum_{v=1}^{n_v} w^{(v)} \sum_{i,j=1}^n (s_{ij} - a_{ij}^{(v)})^2 + \sum_{i,j=1}^n o_{ij} s_{ij} + \sum_{i,j=1}^n \frac{\mu}{2}(s_{ij} + l_{ij})^2 \tag{28}$$

where $o_{ij} = \frac{1}{2} \sum_{v=1}^{n_v} \|b_i^{(v)} - b_j^{(v)}\|_2^2$ is the j-th element of $o_i \in \mathcal{R}^{1 \times n}$, and l_{ij} is the (i, j)-th element of $L = -M + Y_6/\mu$. Equation (28) is independent between different i, and we can address the following problem separately for each i,

$$\min_{s^i \mathbf{1}=1, s^i \geq 0} \sum_{j=1}^n \sum_{v=1}^{n_v} w^{(v)} (s_{ij} - a_{ij}^{(v)})^2 + \sum_{j=1}^n o_{ij} s_{ij} + \sum_{j=1}^n (\frac{\mu}{2} s_{ij}^2 + \mu l_{ij} s_{ij}) \tag{29}$$

Then, s^i in Eq. (29) can be updated by optimizing the following equation:

$$s^{i^{t+1}} = \arg\min_{s^i \mathbf{1}=1, s^i \geq 0} \|s^i + \frac{-2\sum_{v=1}^{n_v} w^{(v)} a_i^{(v)} + o_i + \mu l_i}{2\sum_{v=1}^{n_v} w^{(v)} + \mu}\|_2^2 \tag{30}$$

where $a_i^{(v)}$ and l_i are $1 \times n$ vectors whose j-th elements are $a_{ij}^{(v)}$ and l_{ij}, respectively.

Update $w^{(v)}$ The weight of the v-th view is computed by

$$w^{(v)^{t+1}} = \frac{1}{2\|S - A^{(v)}\|_F} \tag{31}$$

Proof: Motivated by the iteratively reweighted technique in [12, 42], we define an auxiliary problem without $w^{(v)}$ as follows:

$$\min_{A^{(v)}} \sqrt{\|S - A^{(v)}\|_F^2} + \frac{\mu}{2} \|Z^{(v)} - A^{(v)} + \frac{Y_2^{(v)}}{\mu}\|_F^2 \tag{32}$$

Taking the derivative of Eq. (32) with respect to $A^{(v)}$ and setting the derivative to 0, we have

$$\widehat{w}^{(v)} \frac{\partial \|S - A^{(v)}\|_F^2}{\partial A^{(v)}} + \frac{\partial (\frac{\mu}{2} \|Z^{(v)} - A^{(v)} + \frac{Y_2^{(v)}}{\mu}\|_F^2)}{\partial A^{(v)}} = 0 \tag{33}$$

where $\widehat{w}^{(v)} = 1/(2\|S - A^{(v)}\|_F)$. Obviously, Eq. (33) is the same as the derivative process of Eq. (13) with respect to $A^{(v)}$. Thus, $\widehat{w}^{(v)}$ can be considered as $w^{(v)}$ in Eq. (13). Theoretically, to avoid dividing by 0, $\widehat{w}^{(v)}$ can be transformed into

$$w^{(v)^{t+1}} = \frac{1}{2\|S - A^{(v)}\|_F + \delta} \tag{34}$$

where δ is infinitely close to 0. The proof is completed.

Update M Fixing all variables except M, Eq. (12) is equivalent to

$$\lambda_4 \|M\|_{\boxed{k}} + \frac{\mu}{2}\|S - M + \frac{Y_6}{\mu}\|_F^2$$
$$s.t. \, M \geq 0, M = M^T, \tag{35}$$

In view of the nonconvexity of $\|M\|_{\boxed{k}}$ in Eq. (35), we introduce the following theorem:

Theorem 1. [30] Let $M \in \mathcal{R}^{n \times n}$ and $M \succeq 0$. Then,

$$\sum_{i=n-k+1}^{n} \lambda_i(M) = \min_{M} \langle M, Q \rangle$$
$$s.t. \, 0 \preceq Q \preceq I, Tr(Q) = k, \tag{36}$$

where M and Q are symmetric matrices, $M \succeq 0$ represents that M is positive semidefinite, $Q \preceq I$ represents $Q - I \preceq 0$, $Tr(Q)$ represents the sum of the main diagonal elements of Q. We can reformulate $\|M\|_{\boxed{k}}$ as a convex programming problem,

$$\|M\|_{\boxed{k}} = \min_{Q} \langle L_M, Q \rangle$$
$$s.t. \, 0 \preceq Q \preceq I, Tr(Q) = k \tag{37}$$

Therefore, Eq. (35) is equivalent to

$$\lambda_4 \langle Diag(M1) - M, Q \rangle + \frac{\mu}{2}\|S - M + \frac{Y_6}{\mu}\|_F^2$$
$$s.t. \, M \geq 0, M = M^T, 0 \preceq Q \preceq I, Tr(Q) = k \tag{38}$$

Equation (38) can be optimized by solving M and Q alternatively. The specific updating rules are as follows:

(1) Q can be optimized with fixed variable M as

$$Q^{t+1} = \arg\min_{Q} \lambda_4 \langle Diag(M1) - M, Q \rangle$$
$$s.t. \, 0 \preceq Q \preceq I, Tr(Q) = k \tag{39}$$

It can be solved by

$$Q^{t+1} = UU^T \tag{40}$$

where $U \in R^{n \times k}$ consist of k eigenvectors related to the k smallest eigenvalues of $Diag(M1) - M$ [28].

(2) M can be optimized with fixed variable Q as

$$M^{t+1} = \underset{M}{\arg\min} \, \lambda_4 \langle Diag(M1) - M, Q \rangle + \frac{\mu}{2} \| S - M + \frac{Y_6}{\mu} \|_F^2$$
$$s.t. M \geq 0, M = M^T \tag{41}$$

Equation (41) is equivalent to

$$M^{t+1} = \underset{M}{\arg\min} \, \frac{1}{2} \| M - S - \frac{Y_6}{\mu} + \frac{\lambda_4}{\mu}(diag(Q)1^T - Q) \|^2$$
$$s.t. \, M \geq 0, M = M^T \tag{42}$$

Let $D = S + \frac{Y_6}{\mu} - \frac{\lambda_4}{\mu}(diag(Q)1^T - Q)$, according to [30],

$$M^{t+1} = [(\widehat{D} + \widehat{D}^T)/2]_+ \tag{43}$$

where $\widehat{D} = D - Diag(diag(D))$.

Update multipliers $\mu, Y_1^{(v)}, Y_2^{(v)}, Y_3^{(v)}, Y_4^{(v)}, Y_5^{(v)}$ and Y_6 are updated as follows:

$$\mu = \min(\rho\mu, \mu_{max})$$
$$Y_1^{(v)} = Y_1^{(v)} + \mu(X^{(v)} - X^{(v)}Z^{(v)} - E^{(v)})$$
$$Y_2^{(v)} = Y_2^{(v)} + \mu(Z^{(v)} - A^{(v)})$$
$$Y_3^{(v)} = Y_3^{(v)} + \mu(Z^{(v)} - B^{(v)})$$
$$Y_4^{(v)} = Y_4^{(v)} + \mu(Z^{(v)} - C^{(v)})$$
$$Y_5^{(v)} = Y_5^{(v)} + \mu(Z^{(v)} - D^{(v)})$$
$$Y_6 = Y_6 + \mu(S - M) \tag{44}$$

With the help of the alternate optimization scheme, the final S can be obtained and used for clustering. The specific optimization process is summarized in Algorithm 1.

4.1 Complexity Analysis

In our model, there are ten unknown variables (S, Q, M, w, Z, A, B, C,D, E), and it is a nonconvex optimization problem. We alternately update each variable. Let n_v, t_1 and n be the number of views, iterations, and data points, respectively, and we mainly consider computationally expensive operations. The complexity of updating A is $O(n_v n)$. The main complexity of updating B and Z is the matrix inversion, which is $O(n_v n^3)$. To update D (the nuclear norm proximal operator), the main complexity is $O(n_v n^3)$. The main complexity of updating C and S is calculating the Euclidean distance which requires $O(n_v n^2)$. The complexities of updating E is $O(n_v n)$. The complexities of updating Q and M are $O(n^3)$ and $O(n)$, respectively. Since $n_v \ll n$, therefore, the total complexity is $O(t_1 n^3)$.

Since Eq. (10) is nonconvex, it is difficult to ensure that it can converge to a local minimum. Fortunately, most suboptimization problems have a closed-form solution during optimization, and subsequent empirical evidence of the convergence analysis on real datasets also demonstrated that the proposed algorithm has good convergence behavior.

Algorithm 1 Optimization algorithm for CAGL-SGBD

Input: Multiview matrices $\{X^{(1)}, \cdots, X^{(n_v)}\}$, cluster number k, hyperparameter $\lambda_1, \lambda_2, \lambda_3, \lambda_4$.

1: Initialize: $Y_1^{(v)} = Y_2^{(v)} = Y_3^{(v)} = Y_4^{(v)} = Y_5^{(v)} = Y_6 = \mathbf{0}$. S is initialized by weighted-sum rule
   $\frac{1}{n_v}\sum_{v=1}^{n_v} Z^{(v)}$. $A^{(v)} = Z^{(v)}$, $B^{(v)} = Z^{(v)}$, $C^{(v)} = Z^{(v)}$, $D^{(v)} = Z^{(v)}$, $E^{(v)} = \mathbf{0}$, $M = S$, $Z^{(v)}$ is
   initialized by KNN graph. $\mu = 1e^{-2}$, $\rho = 1.2$, $\mu_{max} = 10^6$, threshold value $\epsilon = 10^{-6}$, $maxIter$=100.
2: **while** not converged and $t \leq maxIter$ **do**
3:    **if** $1 \leq v \leq n_v$ **then**
4:        Update $A^{(v)}$ according to Eq. (14);
5:        Update $B^{(v)}$ according to Eq. (16);
6:        Update $C^{(v)}$ according to Eq. (21);
7:        Update $D^{(v)}$ according to Eq. (22);
8:        Update $Z^{(v)}$ according to Eq. (24);
9:        Update $E^{(v)}$ according to Eq. (26);
10:       Update $w^{(v)}$ according to Eq. (31);
11:       Update $Y_1^{(v)}, Y_2^{(v)}, Y_3^{(v)}, Y_4^{(v)}$ and $Y_5^{(v)}$ according to Eq. (44);
12:    **end if**
13:    Update S according to Eq. (30);
14:    Update Q according to Eq. (40);
15:    Update M according to Eq. (43);
16:    Update Y_6 and μ according to Eq. (44);
17:    Check the convergence condition: $\max(\|X^{(v)} - X^{(v)}Z^{(v)} - E^{(v)}\|_\infty, \|Z^{(v)} - A^{(v)}\|_\infty, \|Z^{(v)} -$
       $B^{(v)}\|_\infty, \|Z^{(v)} - C^{(v)}\|_\infty, \|Z^{(v)} - D^{(v)}\|_\infty, \|S - M\|_\infty) \leq \epsilon$;
18: **end while**
19: Apply spectral clustering algortihm to $S = (S + S^T)/2$.
Output: clustering metrics.

---

5 Experiments

5.1 Datasets

(1) MSRC [43] consists of 210 images and 7 classes. For each image, five visual feature vectors are extracted, including color moment CM (24), CENT (254), LBP (256), GIST (512) and HOG (576).

(2) ORL [44] consists of 400 images belonging to 40 distinct subjects with 10 images for each subject. For each image, four feature vectors are extracted, including LBP (59), CENT (254), GIST (512) and HOG (864).

(3) HW [45] consists of 2000 digit images corresponding to 10 classes. For each image, six features are extracted, namely, FOR (76), FAC (216), KAR (64), MOR (6), ZER (47), and PIX (240).

(4) 100 leaves [45] consists of 1600 samples from 100 plant species. For each sample, three features are extracted, namely, 64-D texture histogram, 64-D fine-scale margin, and 64-D shape descriptor.

(5) COIL20 [11] consists of 1440 images and 20 object categories. For each image, three different feature vectors are extracted, namely, the 1024-D intensity feature, 3304-D LBP feature, and 6750 Gabor feature.

(6) BBCSport [45] is a document dataset consisting of 544 documents belonging to 5 classes from the BBC Sport website. In our experiments, two views are used whose dimensions are 3183 and 3203.

Table 1 The default values of the four parameters

Parameter	MSRC	ORL	HW	100 leaves	COIL20	BBCSport
λ_1	100	0.1	0.1	1	0.1	0.1
λ_2	10	5	0.1	1	1	10
λ_3	50	10	1	1	10	5
λ_4	0.05	0.1	0.05	0.001	0.1	0.05

5.2 Comparison Methods and Evaluation Metrics

We compare the proposed method with the following methods: Ncut [46], S-MVSC [44], MCGC [23], MVGL [11], DiMSC [24], CSMSC [47], LMSC [25], MCLES [48], GBS-KO [45], LMVSC [49], CGD [8] and GFSC [26]. For the single-view clustering method Ncut, we employ it for MVC by concatenating the features of each view in a columnwise manner and feed them into Ncut.

Evaluation Metrics: To facilitate evaluation, we use six widely used evaluation metrics to evaluate the clustering performances, including accuracy (ACC), standardized mutual information (NMI), purity, precision, F-score, and adjusted rand index (ARI). For each metric, a larger value indicates a better cluster performance. The running time is also recorded to better reflect the time utilization.

5.3 Experimental Settings

In the experiments, we fix the number of nearest neighbors to 15. For each comparison method, we either use the default parameter settings recommended by the original paper as much as possible in our experiments (if the parameters were provided), or we manually tune them and retain those with the best performances. For our proposed method, the default values of the four parameters on the six datasets when the optimal clustering performances are achieved are listed in Table 1, and the default values will be used in the comparison experiments, parameter analysis, ablation study, visualization, and convergence analysis. Without loss of generality, we run each method 30 times and report the average score and standard deviation.

5.4 Experimental Results

The results are shown in Tables 2, 3, 4, 5, 6 and 7. For each metric, the best and the second-best values are bolded and underlined, respectively. By observing the experimental results, we can obtain the following conclusions.

Our proposed method significantly outperforms all baselines on the MSRC, HW, 100 leaves and COIL20 datasets. Especially on the COIL20 dataset, our method achieves the most perfect clustering results with 100% on all metrics. On the MSRC dataset, our method outperforms the single-view clustering method Ncut by 43.1% in terms of NMI and outperforms the second best MVC method MCLES by 4.52% in terms of NMI. On the HW dataset, in terms of NMI, our method outperforms Ncut by 24.41% and outperforms the second best method MCGC by 2.25%. On the 100 leaves dataset, in terms of NMI, our method outperforms Ncut by 8.68% and outperforms the second best method GBS-KO by 0.47%. On the

Table 2 Clustering performances on the MSRC dataset

Method	ACC	NMI	Purity	Precision	F-score	ARI	Time(s)
Ncut	54.25 (0.00)	43.39 (0.00)	54.28 (0.00)	38.34 (0.00)	39.96 (0.00)	29.81 (0.00)	0.07 (0.05)
S-MVSC	82.19 (0.07)	73.22 (0.03)	83.12 (0.05)	70.51 (0.07)	72.02 (0.05)	67.35 (0.07)	0.05 (0.04)
MCGC	79.52 (0.00)	68.85 (0.00)	79.52 (0.00)	63.52 (0.00)	66.62 (0.00)	60.94 (0.00)	0.37 (0.03)
MVGL	80.47 (0.00)	68.93 (0.00)	80.47 (0.00)	60.97 (0.00)	64.56 (0.00)	58.46 (0.00)	1.99 (0.04)
DiMSC	71.23 (0.00)	62.92 (0.01)	75.57 (0.01)	59.48 (0.01)	61.32 (0.01)	54.86 (0.01)	2.18 (0.05)
CSMSC	82.85 (0.01)	74.85 (0.00)	82.85 (0.01)	72.07 (0.01)	73.39 (0.00)	69.01 (0.00)	2.57 (0.04)
LMSC	73.00 (0.05)	64.39 (0.04)	74.33 (0.03)	60.59 (0.04)	61.83 (0.05)	55.54 (0.05)	3.83 (0.03)
MCLES	88.87 (0.00)	81.97 (0.00)	88.87 (0.00)	77.90 (0.01)	79.38 (0.01)	75.99 (0.01)	5.71(0.04)
GBS-KO	83.33 (0.00)	79.04 (0.00)	83.33 (0.00)	70.25 (0.00)	74.94 (0.00)	70.58 (0.00)	0.24 (0.02)
LMVSC	78.09 (0.00)	68.27 (0.00)	78.09 (0.00)	68.50 (0.00)	67.06 (0.00)	61.62 (0.00)	0.60 (0.01)
CGD	82.47 (0.00)	73.84 (0.00)	82.47 (0.00)	69.03 (0.00)	71.33 (0.00)	66.53 (0.00)	1.86 (0.05)
GFSC	73.52 (0.00)	72.13 (0.00)	79.71 (0.00)	62.92 (0.00)	67.19 (0.00)	61.49 (0.00)	2.71 (0.11)
Ours	**93.19(0.00)**	**86.49(0.00)**	**93.19(0.00)**	**86.19(0.00)**	**86.64(0.00)**	**84.48(0.00)**	8.43 (0.07)

Table 3 Clustering performances on the ORL dataset

Method	ACC	NMI	Purity	Precision	F-score	ARI	Time(s)
Ncut	65.27 (0.02)	79.78 (0.00)	68.05 (0.01)	47.50 (0.02)	51.40 (0.02)	50.19 (0.02)	0.28 (0.05)
S-MVSC	76.03 (0.02)	88.69 (0.00)	79.05 (0.02)	65.00 (0.03)	69.49 (0.02)	68.73 (0.02)	0.11 (0.06)
MCGC	73.93 (0.01)	88.09 (0.00)	78.50 (0.01)	62.25 (0.01)	67.28 (0.01)	66.46 (0.01)	2.04 (0.86)
MVGL	75.25 (0.00)	93.58 (0.00)	82.50 (0.00)	53.64 (0.00)	67.21 (0.00)	66.26 (0.00)	29.73 (0.59)
DiMSC	80.40 (0.02)	91.28 (0.01)	82.91 (0.01)	70.71 (0.02)	74.75 (0.02)	74.13 (0.02)	7.83 (0.12)
CSMSC	84.71 (0.01)	93.66 (0.00)	87.15 (0.01)	77.35 (0.02)	81.07 (0.01)	80.61 (0.01)	9.69 (0.19)
LMSC	75.71 (0.07)	88.32 (0.04)	78.73 (0.07)	64.57 (0.10)	68.54 (0.10)	67.76 (0.10)	9.30 (0.17)
MCLES	84.49 (0.02)	93.62 (0.00)	**87.80(0.01)**	74.01 (0.03)	79.44 (0.02)	78.93 (0.02)	208.78 (1.53)
GBS-KO	83.75 (0.00)	**93.88(0.00)**	86.75 (0.00)	68.75 (0.00)	76.91 (0.00)	76.31 (0.00)	0.46 (0.02)
LMVSC	76.00 (0.00)	91.02 (0.00)	86.50 (0.00)	**81.50(0.00)**	69.54 (0.00)	68.73 (0.00)	1.46 (0.02)
CGD	60.21 (0.02)	81.29 (0.01)	64.70 (0.02)	27.00 (0.04)	39.00 (0.04)	36.92 (0.04)	11.51 (0.10)
GFSC	78.56 (0.02)	92.65 (0.00)	83.92 (0.01)	61.34 (0.06)	71.03 (0.05)	70.25 (0.05)	2.60 (0.02)
Ours	**85.64(0.01)**	93.74 (0.00)	87.56 (0.01)	78.33 (0.02)	**81.67(0.01)**	**81.23(0.01)**	24.43 (0.137)

Table 4 Clustering performances on the HW dataset

Method	ACC	NMI	Purity	Precision	F-score	ARI	Time(s)
Ncut	67.50 (0.00)	71.28 (0.00)	70.60 (0.00)	58.63 (0.00)	63.31 (0.00)	58.90 (0.00)	0.28 (0.05)
S-MVSC	82.17 (0.04)	87.43 (0.02)	84.17 (0.04)	76.19 (0.05)	81.46 (0.04)	79.24 (0.04)	1.21 (0.06)
MCGC	97.15 (0.00)	93.44 (0.00)	97.15 (0.00)	94.32 (0.00)	94.38 (0.00)	93.76 (0.00)	45.22 (0.10)
MVGL	86.00 (0.00)	90.93 (0.00)	88.10 (0.00)	79.53 (0.00)	85.59 (0.00)	83.86 (0.00)	557.10 (5.55)
DiMSC	22.80 (0.00)	10.39 (0.00)	23.98 (0.00)	15.05 (0.00)	15.28 (0.00)	5.76 (0.00)	2377.88 (7.53)
CSMSC	92.28 (0.00)	85.77 (0.00)	92.28 (0.00)	85.25 (0.00)	85.53 (0.00)	83.92 (0.00)	322.29 (7.92)
LMSC	70.41 (0.06)	65.20 (0.04)	71.65 (0.05)	58.32 (0.05)	59.12 (0.05)	54.52 (0.06)	305.43 (7.08)
MCLES	–	–	–	–	–	–	–
GBS-KO	88.10 (0.00)	90.11 (0.00)	88.10 (0.00)	83.27 (0.00)	86.54 (0.00)	84.99 (0.00)	12.68 (0.06)
LMVSC	85.45 (0.00)	80.75 (0.00)	85.45 (0.00)	77.78 (0.00)	77.60 (0.00)	75.11 (0.00)	4.04 (0.02)
CGD	85.44 (0.00)	90.41 (0.00)	87.89 (0.00)	79.01 (0.00)	85.07 (0.00)	83.28 (0.00)	173.62 (0.52)
GFSC	74.73 (0.00)	73.59 (0.00)	76.06 (0.00)	65.25 (0.00)	67.96 (0.00)	64.25 (0.00)	7.00 (0.09)
Ours	**98.20(0.00)**	**95.69(0.00)**	**98.20(0.00)**	**96.38(0.00)**	**96.41(0.00)**	**96.02(0.00)**	1424.75 (23.96)

'–' indicates that the average run time was too long (in excess of 3 h) to obtain the results

Table 5 Clustering performances on the 100 leaves dataset

Method	ACC	NMI	Purity	Precision	F-score	ARI	Time(s)
Ncut	70.45 (0.01)	85.22 (0.00)	72.84 (0.00)	54.75 (0.03)	58.89 (0.02)	58.47 (0.02)	1.20 (0.06)
S-MVSC	81.81 (0.01)	92.15 (0.00)	83.99 (0.01)	72.28 (0.02)	76.40 (0.01)	76.17 (0.01)	0.76 (0.06)
MCGC	74.37 (0.00)	86.41 (0.00)	77.18 (0.00)	59.40 (0.01)	63.06 (0.01)	62.69 (0.01)	23.17 (1.95)
MVGL	69.18 (0.00)	86.78 (0.00)	71.62 (0.00)	17.86 (0.00)	29.33 (0.00)	28.22 (0.00)	95.99 (2.99)
DiMSC	50.21 (0.00)	72.78 (0.00)	52.96 (0.00)	33.00 (0.01)	35.13 (0.00)	34.47 (0.00)	344.23 (5.19)
CSMSC	74.46 (0.01)	87.86 (0.00)	77.13 (0.01)	62.90 (0.01)	66.17 (0.01)	65.83 (0.01)	108.49 (2.83)
LMSC	74.67 (0.01)	87.72 (0.00)	77.03 (0.01)	62.45 (0.02)	65.89 (0.01)	65.55 (0.01)	141.18 (1.86)
MCLES	–	–	–	–	–	–	–
GBS-KO	82.43 (0.00)	93.43 (0.00)	85.12 (0.00)	42.66 (0.00)	57.65 (0.00)	57.10 (0.00)	3.95 (0.04)
LMVSC	65.75 (0.00)	85.03 (0.00)	75.75 (0.00)	63.80 (0.00)	56.55 (0.00)	56.09 (0.00)	2.40 (0.11)
CGD	79.51 (0.01)	92.76 (0.00)	82.36 (0.01)	59.77 (0.05)	70.15 (0.04)	69.82 (0.04)	145.99 (2.77)
GFSC	57.95 (0.01)	78.33 (0.00)	60.99 (0.00)	42.24 (0.01)	45.64 (0.01)	45.09 (0.01)	110.12 (7.488)
Ours	**86.93(0.00)**	**93.90(0.00)**	**88.27(0.00)**	**77.37(0.01)**	**80.87(0.00)**	**80.68(0.00)**	385.71 (4.68)

'_' indicates that the average run time was too long (in excess of 3 h) to obtain the results

Table 6 Clustering performances on the COIL20 dataset

Method	ACC	NMI	Purity	Precision	F-score	ARI	Time(s)
Ncut	79.67 (0.00)	89.91 (0.00)	82.97 (0.00)	72.36 (0.00)	78.05 (0.00)	76.82 (0.00)	0.678 (0.05)
S-MVSC	75.32 (0.01)	84.76 (0.01)	77.74 (0.01)	69.80 (0.02)	72.34 (0.01)	70.85 (0.01)	1.43 (0.05)
MCGC	79.86 (0.00)	89.55 (0.00)	82.98 (0.00)	71.95 (0.00)	77.48 (0.00)	76.21 (0.00)	10.93 (0.08)
MVGL	82.77 (0.00)	92.73 (0.00)	85.00 (0.00)	79.90 (0.00)	83.75 (0.00)	82.86 (0.00)	177.93 (1.25)
DiMSC	77.91 (0.00)	84.33 (0.00)	78.51 (0.01)	73.02 (0.00)	74.25 (0.00)	72.89 (0.00)	231.11 (5.17)
CSMSC	73.54 (0.01)	84.60 (0.00)	77.63 (0.00)	66.60 (0.01)	70.11 (0.00)	68.48 (0.00)	374.64 (26.09)
LMSC	77.17 (0.02)	87.28 (0.01)	80.57 (0.01)	70.10 (0.04)	74.38 (0.03)	72.96 (0.03)	144.85 (1.04)
MCLES	–	–	–	–	–	–	–
GBS-KO	79.09 (0.00)	94.06 (0.00)	84.79 (0.00)	69.38 (0.00)	79.42 (0.00)	78.19 (0.00)	4.91 (0.05)
LMVSC	69.52 (0.00)	68.64 (0.00)	82.38 (0.00)	72.24 (0.00)	60.36 (0.00)	52.72 (0.00)	2.65 (0.11)
CGD	79.65 (0.00)	89.67 (0.00)	82.91 (0.00)	71.61 (0.00)	77.41 (0.00)	76.14 (0.00)	461.36 (3.89)
GFSC	65.56 (0.02)	78.68 (0.01)	69.62 (0.01)	38.33 (0.03)	49.78 (0.02)	46.32 (0.03)	6.65 (2.46)
Ours	**100.00(0.00)**	**100.00(0.00)**	**100.00(0.00)**	**100.00(0.00)**	**100.00(0.00)**	**100.00(0.00)**	392.80(8.82)

'–' indicates that the average run time was too long (in excess of 3 h) to obtain the results

Table 7 Clustering performances on the BBCSport dataset

Method	ACC	NMI	Purity	Precision	F-score	ARI	Time(s)
Ncut	58.63 (0.00)	39.37 (0.00)	61.21 (0.00)	33.52 (0.00)	47.05 (0.00)	20.40 (0.00)	0.11 (0.05)
S-MVSC	87.02 (0.06)	78.16 (0.04)	87.94 (0.04)	**84.15(0.05)**	**84.60(0.04)**	**79.72(0.05)**	27.18 (0.11)
MCGC	40.26 (0.00)	15.81 (0.00)	42.64 (0.00)	25.79 (0.00)	40.34 (0.00)	4.83 (0.00)	2.67 (1.31)
MVGL	38.97 (0.00)	8.90 (0.00)	39.33 (0.00)	24.65 (0.00)	39.19 (0.00)	2.04 (0.00)	6.07 (0.09)
DiMSC	85.91 (0.00)	70.74 (0.00)	85.91 (0.00)	81.23 (0.00)	76.84 (0.00)	70.09 (0.00)	10.27 (0.05)
CSMSC	85.72 (0.00)	72.03 (0.00)	85.72 (0.00)	79.69 (0.00)	81.22 (0.00)	75.19 (0.00)	40.91 (0.54)
LMSC	85.07 (0.00)	74.27 (0.01)	85.07 (0.00)	80.22 (0.01)	81.35 (0.01)	75.40 (0.01)	27.15 (0.21)
MCLES	82.45 (0.03)	73.38 (0.03)	83.49 (0.02)	74.11 (0.07)	79.37 (0.05)	72.17 (0.07)	118.85 (1.48)
GBS-KO	80.69 (0.00)	75.99 (0.00)	84.37 (0.00)	72.71 (0.00)	79.43 (0.00)	72.18 (0.00)	5.68 (0.04)
LMVSC	57.35 (0.00)	35.02 (0.00)	78.12 (0.00)	79.69 (0.00)	44.37 (0.00)	15.16 (0.00)	1.88 (0.02)
CGD	80.69 (0.00)	75.63 (0.00)	84.37 (0.00)	72.17 (0.00)	79.03 (0.00)	71.62 (0.00)	69.18 (0.83)
GFSC	54.12 (0.00)	36.61 (0.00)	54.12 (0.00)	31.71 (0.00)	46.94 (0.00)	17.99 (0.00)	16.85 (0.57)
Ours	**91.27(0.03)**	**81.01(0.04)**	**91.27(0.03)**	80.24 (0.04)	83.77 (0.04)	78.37 (0.05)	39.62 (0.10)

ORL dataset, our method outperforms all baselines except for the NMI, purity and precision, which are 0.14, 0.24, and 3.17% lower than GBS-KO, MCLES and LMVSC, respectively. On the BBCSport dataset, our method achieves the best results in terms of the ACC, NMI and purity. The experimental results show that the proposed method is a promising MVC method.

The performances of MVC methods are not always better than those of the single-view clustering methods. The key problem of MVC is to make full use of the consistency and complementarity information of different views, which is still a challenging problem in practice.

In all the baselines, S-MVSC, MCGC, MVGL, GBS-KO, CGD and GFSC are methods for learning a unified graph/matrix. GBS-KO shows good clustering performances on all six datasets. CGD also shows good performances on all six datasets except for ORL. MCGC shows the second-best best performance on the HW dataset. All three baselines consider the manifold structure of the original data and exploit the complementary information of multiple graphs to generate a unified graph for clustering, but they do not consider the global structure of the data or block representation of the unified graph. Thus, their clustering results are inferior to those of the proposed CAGL-SGBD. GFSC is a method that employs self-weighting to automatically learn a unified graph for all views, and its clustering performance is not good on the 100 leaves, COIL20 and BBCSport datasets. S-MVSC performs well on the BBCSport dataset and shows highly efficient computing performances on all datasets.

Both LMSC and MCLES learn a similarity matrix or cluster indicator matrix based on the unified latent embedding representation in the embedding space. MCLES exhibits the second best clustering performances on the MSRC dataset, and its clustering performances on the ORL and BBCSport datasets are also good, but it cannot be tested on the HW, 100 leaves and COIL20 datasets because it took more than three hours to run. In addition, both LMSC and MCLES may lose some important discriminant information in the process of embedding data from the original space to the embedding space, resulting in lower clustering performances than CAGL-SGBD.

The proposed method CAGL-SGBD is superior to the above unified graph/matrix/representation-based methods. This is mainly because it not only integrates the local manifold structure and global structure information of intraviews through structure graph fusion but also dynamically integrates the complementary information between views by consensus affinity graph learning, and ultimately, it enforces a structured k-block diagonal representation on the learned consensus affinity graph.

Furthermore, the running time of CAGL-SGBD is also within an acceptable range compared to all the baselines.

5.5 Parameter Analysis

Our model has four parameters $\lambda_1, \lambda_2, \lambda_3, \lambda_4$, and we conduct a grid search for them. λ_1 is tuned from {0.001, 0.01, 0.1, 1, 10, 100} on the MSRC dataset, is tuned from {0.001, 0.01, 0.1} on the HW and BBCSport datasets, and is tuned from {0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1} on the ORL, 100 leaves and COIL20 datasets. $\lambda_2, \lambda_3, \lambda_4$ are tuned from {0.1, 0.5, 0.7, 1, 5, 7, 10}, {1, 5, 7, 10, 50, 70, 100} and {0.001, 0.005, 0.007, 0.01, 0.05, 0.07, 0.1}, respectively, on all datasets.

When analyzing one parameter, keep the default values of the other three parameters. Figure 1 takes the ORL, 100 leaves, and COIL20 datasets as examples to show that the variation in NMI values of CAGL-SGBD varies with the four parameters on the three datasets.
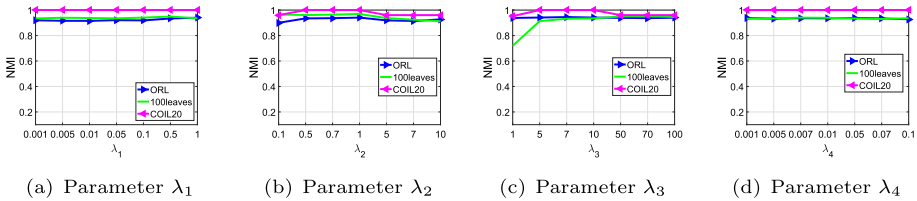
Fig. 1 Variations in NMI versus the four parameters

As shown in Fig. 1, the proposed method is less sensitive to the changes in these four parameters on the three datasets except that it is more sensitive to the changes in λ_3 on the 100 leaves dataset.

5.6 Ablation Study

In this subsection, we conduct an ablation study on the proposed model. Specifically, we learn a consensus affinity graph through the following two approaches: (1) Only consider the structure graph fusion, namely, $\lambda_4 = 0$, simplified as CAGL-SG; (2) Only consider the block representation, namely, $\lambda_1 = \lambda_2 = \lambda_3 = 0$, simplified as CAGL-BD. For the first approach, we have

$$\min_{S, w^{(v)}, Z^{(v)}, E^{(v)}} \sum_{v=1}^{n_v} \{w^{(v)} \|S - Z^{(v)}\|_F^2 + Tr(Z^{(v)} L_S Z^{(v)T}) +$$
$$2\lambda_1 Tr(X^{(v)} L_Z^{(v)} X^{(v)T}) + \lambda_2 \|Z^{(v)}\|_* + \lambda_3 \|E^{(v)}\|_{2,1}\}$$
$$s.t. \ X^{(v)} = X^{(v)} Z^{(v)} + E^{(v)}, Z^{(v)} \mathbf{1} = \mathbf{1}, Z^{(v)} \geq 0, s^i \mathbf{1} = 1, S \geq 0, w^{(v)} \geq 0 \quad (45)$$

For the second approach, we have

$$\min_{S, w^{(v)}} \sum_{v=1}^{n_v} \{w^{(v)} \|S - Z^{(v)}\|_F^2 + Tr(Z^{(v)} L_S Z^{(v)T})\} + \lambda_4 \|S\|_{\boxed{k}}$$
$$s.t. \ s^i \mathbf{1} = 1, S \geq 0, S = S^T, w^{(v)} \geq 0 \quad (46)$$

where $Z^{(v)}$ is initialized by the KNN graph in both Eqs. (45) and (46). Table 8 reports the clustering results of Eqs. (45), (46) and (10) on the MSRC, ORL, HW, 100 leaves and COIL20 datasets.

As shown in Table 8, the clustering performance of Eq. (46) is very poor on some datasets. The model in Eq. (46) directly fuses $\{Z^{(v)}\}_{v=1}^{n_v}$, which are obtained by the KNN graph, into a consistent affinity matrix S and enforces S to have a block diagonal representation. However, the S learned by Eq. (46) cannot capture the local and global structural information of the data well and cannot resist noise; therefore, the clustering performance is not good. Equation (45) obtains better clustering results than Eq. (46) on all datasets, which indicates that the structure graph fusion may play a more important role than block diagonal representation in improving the quality of the consensus affinity matrix. The results of Eq. (10) are better than those of Eqs. (45) and (46), indicating that the joint learning of structure graph fusion and block diagonal representation are beneficial to improving the quality of the consensus affinity matrix. Thus, CAGL-SGBD is superior to CAGL-SG and CAGL-BD.

Table 8 Ablation study of the proposed method

| DataSet	Method	ACC	NMI	Purity	Recall	F-score	ARI
MSRC	Eq. (45)	74.19 (0.01)	77.90 (0.01)	78.47 (0.01)	64.88 (0.01)	71.31 (0.01)	66.15 (0.01)
	Eq. (46)	17.14 (0.00)	10.07 (0.00)	20.00 (0.00)	13.96 (0.00)	24.16 (0.00)	0.21 (0.00)
	Eq. (10)	93.19 (0.00)	86.49 (0.00)	93.19 (0.00)	86.19 (0.00)	86.64 (0.00)	84.48 (0.00)
ORL	Eq. (45)	83.93 (0.01)	92.66 (0.00)	86.00 (0.00)	75.24 (0.01)	78.87 (0.01)	78.36 (0.01)
	Eq. (46)	35.01 (0.01)	65.55 (0.01)	43.26 (0.01)	10.06 (0.01)	17.12 (0.02)	13.80 (0.02)
	Eq. (10)	85.64 (0.01)	93.74 (0.00)	87.56 (0.01)	78.33 (0.02)	81.67 (0.01)	81.23 (0.01)
HW	Eq. (45)	44.85 (0.00)	63.27 (0.00)	53.12 (0.00)	23.06 (0.00)	35.72 (0.00)	24.01 (0.00)
	Eq. (46)	11.35 (0.00)	3.17 (0.00)	11.86 (0.00)	9.97 (0.00)	18.00 (0.00)	0.04 (0.00)
	Eq. (10)	98.20 (0.00)	95.69 (0.00)	98.20 (0.00)	96.38 (0.00)	96.41 (0.00)	96.02 (0.00)
100 leaves	Eq. (45)	86.73 (0.00)	93.80 (0.00)	88.25 (0.00)	76.34 (0.01)	80.31 (0.01)	80.12 (0.01)
	Eq. (46)	13.03 (0.00)	33.09 (0.00)	18.57 (0.00)	1.22 (0.00)	2.40 (0.00)	00.57 (0.00)
	Eq. (10)	86.93 (0.00)	93.90 (0.00)	88.27 (0.00)	77.37 (0.01)	80.87 (0.00)	80.68 (0.00)
COIL.20	Eq. (45)	54.13 (0.00)	79.93 (0.00)	66.82 (0.00)	24.14 (0.00)	37.28 (0.00)	32.11 (0.00)
	Eq. (46)	38.25 (0.01)	46.82 (0.00)	40.73 (0.01)	23.09 (0.01)	27.04 (0.01)	22.55 (0.01)
	Eq. (10)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
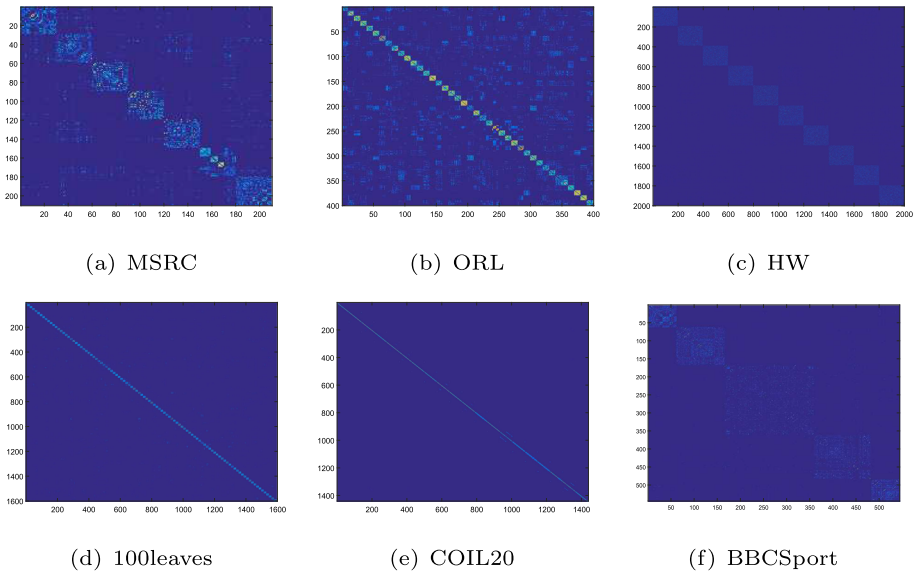
Fig. 2 Visualization results of the consensus affinity matrix learned by CAGL-SGBD on all datasets

5.7 Visualization

The visualization results of the consensus affinity matrix learned by the proposed method on the six datasets are demonstrated in Fig. 2.

The consensus affinity matrix learned by the proposed method on the MSRC, ORL, HW, 100 leaves and BBCSport datasets all exhibit obvious block-diagonal structures, and the number of blocks is equal to the number of corresponding classes. In Fig. 2d, since there are 100 blocks on the 100 leaves dataset, each block is so small that the block structure on the diagonal appears to be a straight line. Although the consensus affinity matrix learned on the COIL20 dataset has no obvious block-diagonal structure, the data points are all concentrated on the diagonal.

We also use t-SNE to visualize the learned consensus affinity matrix, as shown in Fig. 3. It can be seen that data points belonging to the same category are close, while the data points belonging to different categories are far away. On the COIL20 dataset, our proposed method can achieve the most perfect clustering results.

5.8 Convergence Analysis

In this subsection, we conducted convergence analysis to verify the convergence property of the proposed method. We calculate the log objective function value and the primal residual (computed as $\max(\|X^{(v)} - X^{(v)}Z^{(v)} - E^{(v)}\|_\infty, \|Z^{(v)} - A^{(v)}\|_\infty, \|Z^{(v)} - B^{(v)}\|_\infty, \|Z^{(v)} - C^{(v)}\|_\infty, \|Z^{(v)} - D^{(v)}\|_\infty, \|S - M\|_\infty)$) at each iteration. Since our algorithm tends to converge at approximately 50 iterations on the benchmark datasets, we expand the number of iterations to 150 to intuitively demonstrate the convergence of the proposed method at 100 iterations. Figure 4 illustrates the log objective function value and the value of primal residual
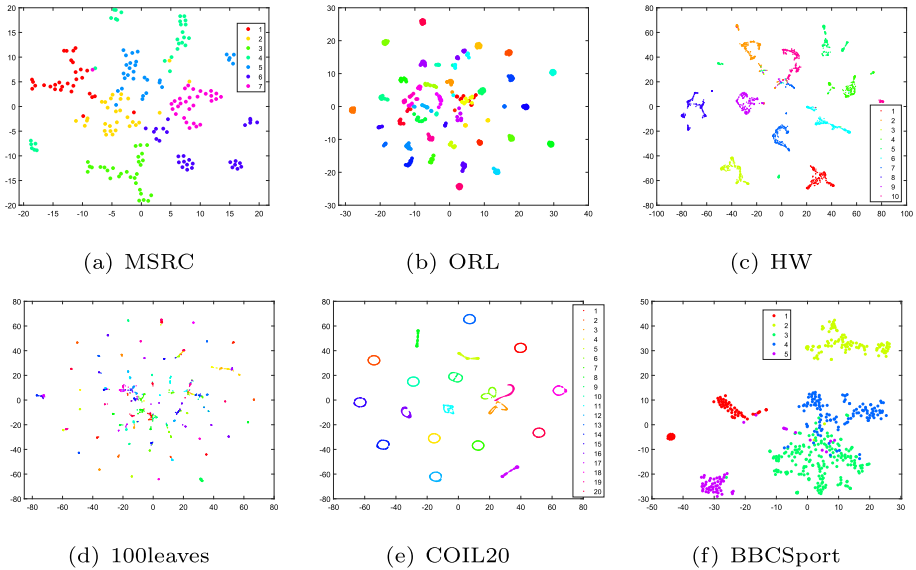
(a) MSRC   (b) ORL   (c) HW

(d) 100leaves   (e) COIL20   (f) BBCSport

Fig. 3 Visualization of the consensus affinity matrix learned by CAGL-SGBD on all datasets using t-SNE

(a) MSRC   (b) ORL   (c) HW

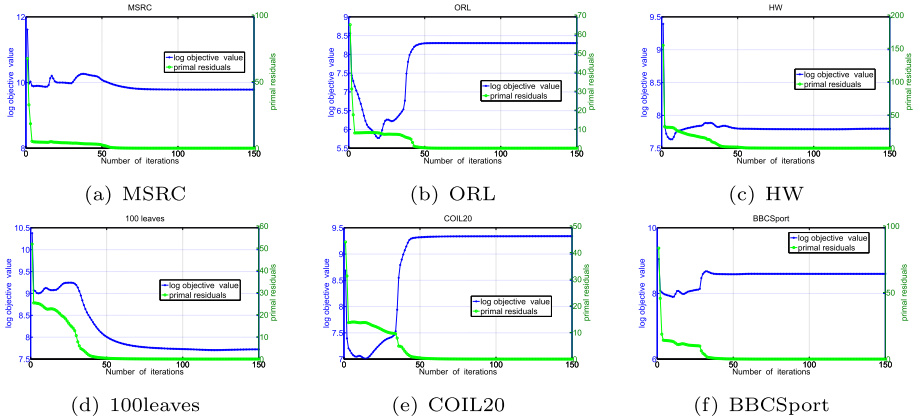(d) 100leaves   (e) COIL20   (f) BBCSport

Fig. 4 Convergence curves of the proposed algorithm

changes with the number of iterations on the six datasets. The results empirically confirm the convergence behavior of the proposed algorithm within 100 iterations.

6 Conclusion

In this paper, we propose a new MVC method CAGL-SGBD. The proposed method can automatically learn the graph structure information of intraviews, the complementary information of interviews and the structured representation of consensus affinity graphs by structure graph fusion, consensus affinity graph learning and k-block diagonal representation, respectively. Extensive experimental results on six benchmark datasets sufficiently show that CAGL-

SGBD is effective, can compete with other advanced MVC methods, and obtains the best performance on the MSRC, HW, 100 leaves and COIL20 datasets. The results of the ablation study show that jointly learning the structure graph fusion and block diagonal representation can greatly improve the clustering performance. The visualization experimental results on the MSRC, ORL, HW, 100 leaves, and BBCSport datasets show that the consensus affinity matrix learned by CAGL-SGBD displays an explicit block-diagonal structure. In the future, we will consider how to address the graph structure and structured representation problem of large-scale multiview data.

Declarations

Conflict of interests 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.

References

1. Wang Q, Chen X, Chen W (2022) Auto-weighted graph regularization and residual compensation for multi-view subspace clustering. Neural Process Lett 54(5):3851–3871
2. Yin Q, Zhang J, Wu S, Li H (2019) Multi-view clustering via joint feature selection and partially constrained cluster label learning. Pattern Recogn 93:380–391
3. Tang C, Zheng X, Liu X, Zhang W, Zhang J, Xiong J, Wang L (2022) Cross-view locality preserved diversity and consensus learning for multi-view unsupervised feature selection. IEEE Trans Knowl Data Eng 34(10):4705–4716
4. Zhang P, Li T, Wang G et al (2021) Multi-source information fusion based on rough set theory: a review. Inf Fusion 68:85–117
5. Dong W, Wu X, Xu T (2022) Multi-view subspace clustering via joint latent representations. Neural Process Lett 54(3):1879–1901
6. Liu T, Martin G, Zhu Y, Peng L, Li L (2020) Joint robust multi-view spectral clustering. Neural Process Lett 52:1843–1862
7. Tang C, Zhu X, Liu X, Li M, Wang P, Zhang C, Wang L (2019) Learning a joint affinity graph for multiview subspace clustering. IEEE Trans Multimed 21(7):1724–1736
8. Tang C, Liu X, Zhu X, Zhu E, Luo Z,Wang L, Gao W (2020) CGD: multi-view clustering via cross-view graph diffusion. In: AAAI, 2020, 34(04): pp 5924-5931
9. Hu Z, Nie F, Wang R, Li X (2020) Multi-view spectral clustering via integrating nonnegative embedding and spectral embedding. Inf Fusion 55:251–259
10. Wang H, Yang Y, Liu B (2020) GMC: graph-based multi-view clustering. IEEE Trans Knowl Data Eng 32(6):1116–1129
11. Zhan K, Zhang C, Guan J, Wang J (2017) Graph learning for multiview clustering. IEEE Trans Cybern 48(10):2887–2895
12. Ren Z, Yang S, Sun Q, Wang T (2020) Consensus affinity graph learning for multiple kernel clustering. IEEE Trans Cybern 51(6):3273–3284
13. Ren Z, Lei H, Sun Q, Yang C (2020) Simultaneous learning coefficient matrix and affinity graph for multiple kernel clustering. Inf Sci 547:289–306

14. Huang J, Nie F, Huang H (2013) Spectral rotation versus K-means in spectral clustering. In: Proceedings of the AAAI Conference on Artificial Intelligence, 27(1): 431-437

15. Kang Z, Lu Y, Su Y, Li C, Xu Z (2019) Similarity Learning via Kernel Preserving Embedding. In: Proceedings of the AAAI conference on artificial intelligence, 33(01): 4057-4064

16. Nie F, Cai G, Li J, Li X (2017) Auto-weighted multi-view learning for image clustering and semi-supervised classification. IEEE Trans Image Process 27(3):1501–1511

17. Huang Y, Xiao Q, Du S, Yu Y (2022) Multi-view clustering based on low-rank representation and adaptive graph learning. Neural Process Lett 54(1):265–283

18. Nie F, Cai G, Li X (2017) Multi-view clustering and semi-supervised classification with adaptive neighbours. In: 31st AAAI conference on artificial intelligence, 31(1): 2408-2414

19. Li CG, Vidal R (2015) Structured sparse subspace clustering: a unified optimization framework. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 277-286

20. Yan H, Liu S, PS Y (2019) From joint feature selection and self-representation learning to robust multi-view subspace clustering. In: 2019 IEEE international conference on data mining (ICDM), pp 1414-1419

21. Kang Z, Peng C, Cheng Q (2017) Clustering with adaptive manifold structure learning. In: 2017 IEEE 33rd International Conference on Data Engineering (ICDE), pp 79-82

22. Kang Z, Peng C, Cheng Q, Liu X, Peng X, Xu Z, Tian L (2021) Structured graph learning for clustering and semi-supervised classification. Pattern Recogn 110:107627

23. Zhan K, Nie F, Wang J, Yang Y (2019) Multiview consensus graph clustering. IEEE Trans Image Process 28(3):1261–1270

24. Cao X, Zhang C, Fu H, Liu S, Zhang H (2015) Diversity induced multi-view subspace clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 586–594

25. Zhang C, Hu Q, Fu H, Zhu P, Cao X (2017) Latent multi-view subspace clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4279–4287

26. Kang Z, Shi G, Huang S, Chen W, Pu X, Zhou JT, Xu Z (2020) Multi-graph fusion for multi-view spectral clustering. Knowl Based Syst 189:105102

27. Liu M, Wang Y, Sun J, Ji Z (2020) Structured block diagonal representation for subspace clustering. Appl Intell 50(8):2523–2536

28. Yin M, Liu W, Li M, Jin T, Ji R (2021) Cauchy loss induced block diagonal representation for robust multi-view subspace clustering. Neurocomputing 427:84–95

29. Wang L, Huang J, Yin M, Cai R, Hao Z (2020) Block diagonal representation learning for robust subspace clustering. Inf Sci 526:54–67

30. Lu C, Feng J, Lin Z, Mei T, Yan S (2018) Subspace clustering by block diagonal representation. IEEE Trans Pattern Anal Mach Intell 41(2):487–501

31. Zhang H, Wu D, Nie F, Wang R, Li X (2021) Multilevel projections with adaptive neighbor graph for unsupervised multi-view feature selection. Inf Fusion 70:129–140

32. Gui Z, Yang J, Xie Z (2022) Learning an enhanced consensus representation for multi-view clustering via latent representation correlation preserving. Knowl Based Syst 253:109479

33. Wang Q, Jiang X, Chen M, Li X (2021) Autoweighted multiview feature selection with graph optimization. IEEE Trans Cybern 52(12):12966–12977

34. Wang R, Nie F, Wang Z, Hu H, Li X (2019) Parameter-free weighted multi-view projected clustering with structured graph learning. IEEE Trans Knowl Data Eng 32(10):2014–2025

35. Nie F, Wu D, Wang R, Li X (2020) Self-weighted clustering with adaptive neighbors. IEEE Trans Neural Netw Learn Syst 31(9):3428–3441

36. Han N, Wu J, Liang Y, Fang X, Wong WK, Teng S (2018) Low-rank and sparse embedding for dimensionality reduction. Neural Netw 108:202–216

37. Liu G, Lin Z, Yan S, Sun J, Yu Y, Ma Y (2012) Robust recovery of subspace structures by low-rank representation. IEEE Trans Pattern Anal Mach Intell 35(1):171–184

38. Abhadiomhen SE, Wang ZY, Shen XJ (2022) Coupled low rank representation and subspace clustering. Appl Intell 52(1):530–546

39. Wang Z, Abhadiomhen SE, Liu Z, Shen X, Gao W, Li S (2021) Multi-view intrinsic low-rank representation for robust face recognition and clustering. IET Image Proc 15(14):3573–3584

40. Abhadiomhen SE, Wang Z, Shen X, Fan J (2021) Multiview common subspace clustering via coupled low rank representation. ACM Trans Intel Syst Technol (TIST) 12(4):1–25

41. Cai JF, Candès EJ, Shen Z (2010) A singular value thresholding algorithm for matrix completion. SIAM J OPTIMIZ 20(4):1956–1982

42. Nie F, Li J, Li X (2017) Self-weighted Multiview Clustering with Multiple Graphs. In: IJCAI, pp 2564-2570

43. Nie F, Li J, Li X (2016) Parameter-free auto-weighted multiple graph learning: a framework for multiview clustering and semi-supervised classification. In: IJCAI, pp 1881-1887

44. Hu Z, Nie F, Chang W, Hao S, Wang R, Li X (2020) Multi-view spectral clustering via sparse graph learning. Neurocomputing 384:1–10
45. Wang H, Yang Y, Liu B, Fujita H (2019) A study of graph-based system for multi-view clustering. Knowl Based Syst 163:1009–1019
46. Shi J, Malik J (2000) Normalized cuts and image segmentation. IEEE Trans Pattern Anal Mach Intell 22(8):888–905
47. Luo S, Zhang C, Zhang W, Cao X (2018) Consistent and specific multi-view subspace clustering. In: AAAI, 32(1):3730-3737
48. Chen MS, Huang L, Wang C, Huang D (2020) Multi-view clustering in latent embedding space. In: AAAI 34(04):3513–3520
49. Kang Z, Zhou W, Zhao Z, Shao J, Han M, Xu Z (2020) Large-scale multi-view subspace clustering in linear time. In: Proceedings of the AAAI conference on artificial intelligence, 34(04): 4412-4419