Augmented Sparse Representation for Incomplete Multiview Clustering

Jie Chen, *Member, IEEE*, Shengxiang Yang, *Senior Member, IEEE*, Xi Peng, *Member, IEEE*, Dezhong Peng, *Member, IEEE*, and Zhu Wang

***Abstract*— Incomplete multiview data are collected from multiple sources or characterized by multiple modalities, where the features of some samples or some views may be missing. Incomplete multiview clustering (IMVC) aims to partition the data into different groups by taking full advantage of the complementary information from multiple incomplete views. Most existing methods based on matrix factorization or subspace learning attempt to recover the missing views or perform imputation of the missing features to improve clustering performance. However, this problem is intractable due to a lack of prior knowledge, e.g., label information or data distribution, especially when the missing views or features are completely damaged. In this article, we proposed an augmented sparse representation (ASR) method for IMVC. We first introduce a discriminative sparse representation learning (DSRL) model, which learns the sparse representations of multiple views as applied to measure the similarity of the existing features. The DSRL model explores complementary and consistent information by integrating the sparse regularization item and a consensus regularization item, respectively. Simultaneously, it learns a discriminative dictionary from the original samples. The sparsity constrained optimization problem in the DSRL model can be efficiently solved by the alternating direction method of multipliers (ADMM). Then, we present a similarity fusion scheme, namely, a sparsity augmented fusion of sparse representations, to obtain a sparsity augmented similarity matrix across different views for spectral clustering. Experimental results on several datasets demonstrate the effectiveness of the proposed ASR method for IMVC.**

***Index Terms*— Incomplete multiview data, multiview clustering (MVC), similarity fusion, sparse representation.**

NOMENCLATURE

Symbol	Description
k	Dimensionality of subspace.
d	Dimensionality of sample.
n	Number of samples or features.
N	Number of incomplete views.
d_v	Dimensionality of features in the vth view.
n_v	Number of existing features in the vth view.
$\mathbf{X} \in \mathbb{R}^{d\times n}$	Matrix containing the n samples.
$\mathbf{Z} \in \mathbb{R}^{n\times n}$	Sparse coefficient matrix.
$\mathbf{X}^{(v)} \in \mathbb{R}^{d_v\times n}$	Matrix containing the n features in the vth view.
$\mathbf{X}_c^{(v)} \in \mathbb{R}^{d_v\times n_v}$	Matrix containing the n_v features in the vth view.
$\mathbf{D}^{(v)} \in \mathbb{R}^{d_v\times n_v}$	Dictionary in the vth view.
$\mathbf{Z}^{(v)} \in \mathbb{R}^{n_v\times n_v}$	Sparse coefficient matrix in the vth view.
$\mathbf{M}^{(v)} \in \mathbb{R}^{n\times n}$	Diagonal indicator matrix.
$\|\mathbf{X}\|_0$	Number of nonzero elements in $\mathbf{X}$.
$\|\mathbf{X}\|_F$	Frobenius norm of $\mathbf{X}$.
$\|\mathbf{X}\|_{\max}$	$\|\mathbf{X}\|_{\max} = \max_{1\le i\le d, 1\le j\le n}\|\mathbf{x}_{ij}\|$.
$\mathbf{X}^T$	Transpose of $\mathbf{X}$.
$\mathbf{X}^{-1}$	Inverse of $\mathbf{X}$.
$diag(\mathbf{X})$	Vector containing the n diagonal elements of $\mathbf{X}$.
$tr(\mathbf{X})$	Trace of $\mathbf{X}$.

Manuscript received 5 October 2021; revised 11 April 2022 and 23 June 2022; accepted 22 August 2022. Date of publication 7 September 2022; date of current version 1 March 2024. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61303015, in part by the National Key Project under Grant GJXM92579, in part by the Sichuan Science and Technology Program under Grant 2021YJ0078, and in part by the National Key Research and Development Program of China under Grant 2018YFC0830300. *(Corresponding author: Zhu Wang.)*

Jie Chen, Xi Peng, and Dezhong Peng are with the College of Computer Science, Sichuan University, Chengdu 610065, China (e-mail: chenjie2010@scu.edu.cn; pengx.gm@gmail.com; pengdz@scu.edu.cn).

Shengxiang Yang is with the School of Computer Science and Informatics, De Montfort University, Leicester LE1 9BH, U.K. (e-mail: syang@dmu.ac.uk).

Zhu Wang is with the Law School, Sichuan University, Chengdu 610065, China (e-mail: wangzhu@scu.edu.cn).

Digital Object Identifier 10.1109/TNNLS.2022.3201699

I. INTRODUCTION

WITH the advancement of information acquisition technology, data are generally collected from multiple sources or characterized by multiple modalities. For example, news may be reported in different languages, and a facial image can be described using different types of features [4], [8], e.g., eigenfaces, fisherfaces, and laplacianfaces. Such data are referred to as multiview data. Each view of multiview data usually has its own individual properties. In multiview data, consistent and complementary information is provided among different views. Multiview clustering (MVC) optimally integrates the information from different views to improve clustering performance compared with obtaining information from an individual view [6], [31], [32], [37].

A number of MVC approaches have been proposed to exploit the intrinsic structures of multiview data in recent years [17], [19], [21], [34], [45]. These approaches usually learn a common data representation, a common indicator matrix, or a joint graph compatible across multiple views for MVC. Chen *et al.* [7] presented the multiview low-rank representation (LRR) method to discover the low-dimensional structures of multiple views under a symmetric constraint.

Nie *et al.* [29] presented a scalable and parameter-free graph fusion framework for MVC. However, most studies often assume that all the features of samples are available and tightly aligned in multiple views. In practice, collecting complete views of all the samples is an intractable problem due to various reasons, e.g., the high cost of data collection and exceptions on acquisition equipment. As a result, the features of some samples do not appear in all views. Such data are called incomplete multiview data. This brings about difficulty for most existing MVC methods to exploit consistent and complementary information from incomplete views. Moreover, the diversity of different clusters across multiple views cannot be encouraged directly because of the missing sample features in some views. Consequently, reducing the negative impact of missing features on the performance of incomplete multiview clustering (IMVC) is a challenging problem.

Recently, some IMVC methods have been proposed to alleviate the above problem. From the perspective of clustering techniques, the existing IMVC methods are mainly divided into five categories, subspace learning-based methods [41], [43], nonnegative matrix factorization (NMF)-based methods [15], [16], graph-learning-based methods [20], [39], multiple kernel-based methods [23], [24], and deep-learning-based methods [21], [38], [42], [44], [46]. Wen *et al.* [41] present an extension of an LRR model that incorporates feature-space-based missing-view inferring and manifold space-based similarity graph learning. NMF aims to factorize the data matrix into two nonnegative matrices, which are considered to be a basis matrix and a latent feature matrix in MVC. Hu and Chen [15] combine an $l_{2,1}$-norm regularized regression imposed on the feature matrix of individual views and NMF to reduce the influence of incomplete views. Liu *et al.* [23] propose an IMVC method to learn a consensus clustering matrix by imputing each incomplete base matrix generated by the incomplete views, where the process of completing the missing features is replaced with imputing the incomplete kernel matrices. Xu *et al.* [42] integrated elementwise reconstruction into a generative adversarial network to infer missing data in multiview data. Although these methods can obtain promising results, there are still some limitations and drawbacks in IMVC. First, it may be difficult for the improved LRR model to learn a complete low-dimensional representation if the features of some samples are missing in a few views. Second, the nonnegative constraint of the latent feature matrix may be insufficient to characterize the incomplete multiview data using the learned basis matrix, although NMF has straightforward interpretability for the feature matrix. Finally, imputation is meaningful and beneficial for IMVC when some features of multiview data are contaminated with small corruptions or noise. However, incomplete multiview data are not well recovered if all features of some samples are missing in some views. These issues indicate that IMVC still faces significant challenges.

The research in [24] argues that IMVC is essentially a task of information fusion, which can be performed at different levels, e.g., a raw data level, feature level, or decision level. An imputation strategy that fills the missing instances into incomplete views is usually adopted at the raw data level. The latent feature matrix shared by the distinct basis matrices of the individual views can be obtained from a weighted NMF model at the feature level [15]. The imputation to an incomplete similarity matrix is performed at the decision level. Most imputations are considered to be a preprocessing step in information fusion. However, imputation with respect to missing features of the incomplete views or insufficient data entries of the incomplete similarity matrix could adversely affect the clustering performance of multiple views due to the effect of low quality [24]. Essentially, it is the critical step of IMVC to determine the whole similarity matrix, each of whose elements is used to measure the relationship of all the samples. As a result, effective information fusion of incomplete similarity matrices has great significance to IMVC.

The IMVC problem can be regarded as determining a finite set of clusters with respect to a given similarity measure. From the perspective of clustering, similar samples are grouped in the same cluster, and dissimilar samples are divided into different clusters according to a complete similarity matrix. This similarity matrix is sparse, as most elements are zeros. Motivated by recent progress in sparse representation [1], [9], [11], a number of subspace clustering methods are proposed to address the clustering problem of high-dimensional data in a single view [12], [28]. These methods rely on the self-expressiveness property of high-dimensional data by representing each data sample as a sparse linear combination of other samples. However, these methods are insufficient to cope with the challenges faced with incomplete multiview data.

In this article, we present an augmented sparse representation (ASR) method for IMVC. The flowchart of the proposed method is demonstrated in Fig. 1. Specifically, we first introduce a discriminative sparse representation learning (DSRL) model, which makes full use of the self-expressiveness property to learn the sparse representations of multiple views. We integrate the sparse regularization item and a consensus regularization item into the DSRL model to explore the complementary and consistent information of different views. Simultaneously, the DSRL model learns a discriminative dictionary from the original samples. Sparse representations of multiple views are adopted to measure the similarity of the existing features. Then, we present a similarity fusion scheme, namely, a sparsity augmented fusion of sparse representations, to obtain a sparsity augmented similarity matrix across different views, which preserves the similarity between each pair of samples, for spectral clustering. In addition, we show the convergence analysis and complexity analysis of the ASR method.

Our major contributions are summarized as follows.

1) The DSRL model that integrates the sparse regularization item and a consensus regularization item to explore the complementary and consistent information across multiple views is presented.
2) An alternating iterative optimization algorithm to learn a discriminative dictionary for sparse representation and simultaneously obtain sparse representations of multiple views in the DSRL model is proposed.
3) A sparsity augmented fusion of sparse representations to obtain a sparsity augmented similarity matrix across different views under a theoretical convergence guarantee is presented.

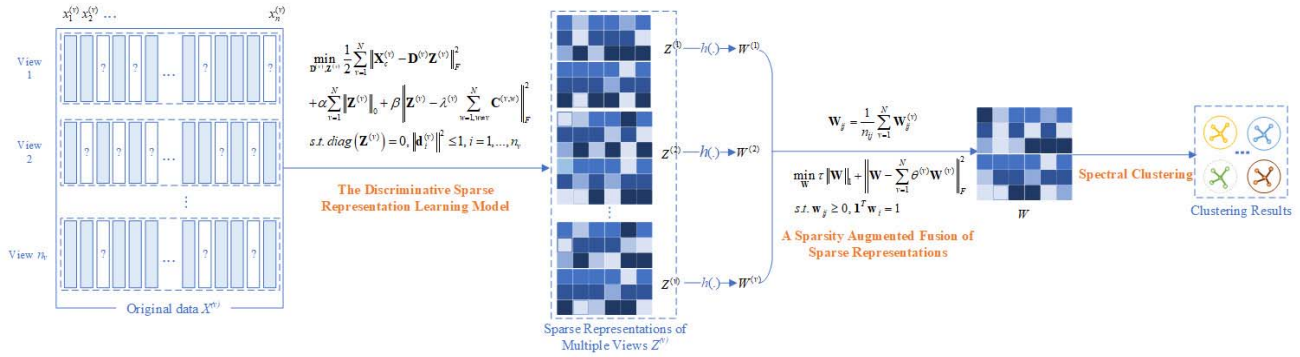

Fig. 1. Flowchart of the ASR method. In the vth view, $\mathbf{X}_c^{(v)}$ consists of the existing features, $\mathbf{D}^{(v)}$ is a dictionary, $\mathbf{Z}^{(v)}$ is a sparse coefficient matrix, and $\mathbf{W}^{(v)}$ is used to measure the similarities among the existing features. Finally, $\mathbf{W}$ is a sparsity augmented similarity matrix across multiple views.

4) Extensive experimental results show that the proposed ASR method achieves considerable improvement over the state-of-the-art IMVC methods.

The remainder of this article is organized as follows. In Section II, we give a brief review of related work. In Section III, we present the proposed ASR method and its theoretical analysis. Extensive experimental results are reported in Section IV. Finally, the conclusion is given in Section V.

II. Related Work

In this section, we briefly review the most related work, i.e., sparse representation techniques and IMVC methods. For consistency, the descriptions of the symbols are summarized in Nomenclature section.

A. Problem Formulation

Given a set of incomplete multiview data $\{\mathbf{X}^{(v)} \in \mathbb{R}^{d_v \times n}\}$ ($v \in \{1, 2, \ldots, N\}$) with n samples, N views, and c clusters, $\mathbf{X}^{(v)}$ is the vth view of the multiview data. Each sample has an individual feature in each view, and all the features of each sample are strictly aligned in multiple views. Each view $\mathbf{X}^{(v)}$ has a total of n features of samples, i.e., $\mathbf{X}^{(v)} = [\mathbf{x}_1^{(v)}, \mathbf{x}_2^{(v)}, \ldots, \mathbf{x}_n^{(v)}]$, where all the missing features of the samples are filled with zeros. Specifically, a diagonal indicator matrix $\mathbf{M}^{(v)} \in \mathbb{R}^{n \times n}$ is defined as

$$\mathbf{M}_{ii}^{(v)} = \begin{cases} 1, & \text{the existing feature } \mathbf{x}_i^{(v)} \text{ in the } v\text{th view} \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

IMVC aims to group n samples into c clusters by integrating all N incomplete views.

B. Sparse Representation Techniques

Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$ be a matrix, and each column of $\mathbf{X}$ represents a sample. By virtue of the self-expressiveness property of high-dimensional data, a sample $\mathbf{x}_i \in \mathbb{R}^d$ ($1 \le i \le n$) is represented by the linear combination of the elements of $\mathbf{X}$

$$\mathbf{x}_i = \mathbf{X}\mathbf{z}_i + \mathbf{e} \tag{2}$$

Algorithm 1 Vector Projection Algorithm [11]

1: **Input:** A vector $\mathbf{v} \in \mathbb{R}^n$ and a parameter $\sigma > 0$
2: Sort $\mathbf{v}$ into $\mathbf{q}$: $\mathbf{q}_1 \ge \mathbf{q}_2 \ge \cdots \ge \mathbf{q}_n$,
3: Search $m = \max\left\{j : \mathbf{q}_j - \frac{1}{j}\left(\sum_{i=1}^{j} \mathbf{q}_i - \sigma\right) > 0, j \in [n]\right\}$,
4: Let $\varphi = \frac{1}{m}\left(\sum_{i=1}^{m} \mathbf{q}_i - \sigma\right)$,
5: **Output:** $\mathbf{r}$ where $\mathbf{r}_i = max(\mathbf{v}_i - \varphi, 0), i \in [n]$.

where $\mathbf{z}_i \in \mathbb{R}^n$ is a coefficient vector, and $\mathbf{e} \in \mathbb{R}^d$ is an error term. Inspired by the advances in the l_0-norm and l_1-norm techniques, $\mathbf{z}_i$ is expected to contain a small number of significant coefficients [1], [9], [11]. For instance, Elhamifar and Vidal [12] proposed a sparse subspace clustering (SSC) method to exploit the low-dimensional subspace structures of high-dimensional data. The SSC method considers a sparsity constraint in the following optimization problem:

$$\min_{\mathbf{Z},\mathbf{E}} \ \|\mathbf{Z}\|_0 + \alpha\|\mathbf{E}\|_F^2 \quad \text{s.t. } \mathbf{X} = \mathbf{X}\mathbf{Z} - \mathbf{E}, \ \text{diag}(\mathbf{Z}) = 0 \tag{3}$$

where $\mathbf{Z} \in \mathbb{R}^{n \times n}$ is a sparse coefficient matrix, and $\mathbf{E} \in \mathbb{R}^{d \times n}$ is an error matrix. In particular, $\mathbf{Z}$ is regarded as a sparse representation of $\mathbf{X}$, which is used to measure the similarity among samples. For example, $\mathbf{z}_{ij}$ represents the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$. This problem can be solved using an alternating direction method of multipliers (ADMM) framework [2]. After $\mathbf{Z}$ is obtained, spectral clustering algorithms, e.g., Ncuts, are applied to obtain the final clustering result [25], [33].

Duchi *et al.* [11] proposed an efficient projection algorithm, which performs Euclidean projection onto the positive simplex, onto the l_1-Ball to obtain high degrees of sparsity of the high-dimensional data. Given a vector $\mathbf{v} \in \mathbb{R}^n$, a sparse vector $\mathbf{r} \in \mathbb{R}^n$ is achieved by solving the following problem:

$$\min \ \|\mathbf{r} - \mathbf{v}\|_F^2 \quad \text{s.t. } \sum_{i=1}^{n} \mathbf{r}_i = \sigma, \quad \mathbf{r}_i \ge 0 \tag{4}$$

where σ is a scalar. Problem (4) has a closed-form solution, which can be efficiently solved by Algorithm 1 [11]. The computational complexity of Algorithm 1 is $\mathcal{O}(n \log(n))$.

C. IMVC Methods

NMF essentially aims to find a compressed approximation of the original data matrix. Given the matrix $\mathbf{X}$, it can be

factorized into two nonnegative matrices, i.e., $\mathbf{U} \in \mathbb{R}^{d\times k}$ and $\mathbf{V} \in \mathbb{R}^{n\times k}$, where $\mathbf{U}$ and $\mathbf{V}$ are a basis matrix and a latent feature matrix, respectively. The NMF optimization problem is formulated as follows [3]:

$$\min_{\mathbf{U},\mathbf{V}} \left\|\mathbf{X}-\mathbf{U}\mathbf{V}^T\right\|_F^2 \quad \text{s.t. } \mathbf{U}\geq 0, \quad \mathbf{V}\geq 0. \tag{5}$$

Several NMF-based methods have been proposed to learn a common feature matrix by introducing $l_{2,1}$-norm regularized regression [15], [16]. For example, the optimization problem of the weighted semi-NMF for IMVC is formulated as

$$\begin{aligned}\min_{\mathbf{U}^{(v)},\mathbf{V},\mathbf{B}^{(v)}} \quad & \sum_{v=1}^{N}\left(\left\|\left(\mathbf{X}^{(v)}-\mathbf{U}^{(v)}\mathbf{V}^T\right)\mathbf{M}^{(v)}\right\|_F^2\right) \\ & +\alpha\sum_{v=1}^{N}\left(\left\|\mathbf{B}^{(v)^T}\mathbf{U}^{(v)}-\mathbf{I}\right\|_F^2+\beta\left\|\mathbf{B}^{(v)}\right\|_{2,1}\right) \\ \text{s.t. } & \mathbf{V}\geq 0\end{aligned} \tag{6}$$

where α and β are two parameters, and $\mathbf{B}^{(v)} \in \mathbb{R}^{d_v\times k}$ is a regression coefficient matrix for the vth view [15]. A novel alternating iteration procedure on the variables $\{\mathbf{U}^{(v)}\}_{v=1}^N$, $\{\mathbf{B}^{(v)}\}_{v=1}^N$, and $\mathbf{V}$ can be applied to solve problem (6). The common latent feature matrix $\mathbf{V}$ is calculated by solving the above optimization problems [15]. Finally, the final clustering indicators can be obtained by applying k-means on $\mathbf{V}$ [18]. These methods avoid the requirement of imputing missing features with the average.

III. Augmented Sparse Representation for IMVC

In this section, we present the proposed ASR algorithm, which learns a sparsity augmented similarity matrix across different views for IMVC. The proposed ASR algorithm is mainly composed of two submodels, DSRL and sparsity augmented fusion of sparse representations for multiple views.

A. Discriminative Sparse Representation Learning for Incomplete Multiple Views

Considering the indicator matrix $\mathbf{M}^{(v)}$ given in the vth view of incomplete multiview data, the features of the vth view can be represented by $\mathbf{X}^{(v)}\mathbf{M}^{(v)}$. The missing features are replaced with zeros in $\mathbf{X}^{(v)}\mathbf{M}^{(v)}$. In particular, the sample is regarded as an outlier if all the features are missing in all the views. Suppose $\mathbf{X}_c^{(v)} \in \mathbb{R}^{d_v\times n_v}$ consists of the existing features in the vth view, where n_v is the number of the existing features. This indicates that the columns of $\mathbf{X}^{(v)}\mathbf{M}^{(v)}$ containing all zeros are removed.

For incomplete multiview data, we first explore the complementary information across different views by learning a sparse representation while ignoring the underlying information of the missing features. Specifically, a sparse representation model that contains a sparse constraint is formulated as

$$\begin{aligned}\min_{\mathbf{Z}^{(v)}} \quad & \frac{1}{2}\sum_{v=1}^{N}\left\|\mathbf{X}_c^{(v)}-\mathbf{X}_c^{(v)}\mathbf{Z}^{(v)}\right\|_F^2+\alpha\sum_{v=1}^{N}\left\|\mathbf{Z}^{(v)}\right\|_0 \\ \text{s.t. } & \operatorname{diag}\left(\mathbf{Z}^{(v)}\right)=0\end{aligned} \tag{7}$$

where α is a trade-off parameter. The diagonal constraint $\operatorname{diag}(\mathbf{Z}^{(v)})=0$ is used to prevent the trial solutions of sparse representation in problem (7). In addition, $\mathbf{Z}^{(v)}$ is a coefficient matrix, which is considered to be a sparse representation for the vth view, where $\mathbf{z}_{ij}^{(v)}$ is used to measure the similarity between each pair of features $\mathbf{x}_i^v$ and $\mathbf{x}_j^v$ in the vth view. The sparse regularization item $\|\mathbf{Z}^{(v)}\|_0$ allows us to capture a local structure of incomplete multiview data. This enables the sparse representation model to seek the relationship of the existing features. Consequently, the complementary information across different views contained in $\mathbf{Z}^{(v)}(v=1,2,\ldots,N)$ can be effectively exploited in the sparse representation model.

Then, we consider the consistent information across different views of incomplete multiview data. The indices of columns of the missing features in a view may be completely different from that of the missing features in another view. The size of $\mathbf{Z}^{(v)}$ may be different from others in multiple views since it characterizes the relationship among the existing features. Hence, we design a similarity alignment rule for different sparse representations of incomplete multiple views. For clarity, we suppose that $\mathbf{W}^{(v)} \in \mathbb{R}^{n\times n}$ represents the similarity of the complete features in the vth view. Specifically, $\mathbf{W}_{ij}^{(v)}$ represents the similarity between $\mathbf{x}_i^{(v)}$ and $\mathbf{x}_j^{(v)}$. A mapping function of the rule $h(\cdot)$ is defined as

$$\mathbf{W}^{(v)}=h\left(\mathbf{Z}^{(v)},\mathbf{M}^{(v)}\right). \tag{8}$$

Here, $h(\cdot)$ follows the rule that all the elements of $\mathbf{Z}^{(v)}$ are filled into the corresponding entries of $\mathbf{W}^{(v)}$, and the other entries of $\mathbf{W}^{(v)}$ that are used to measure similarities between the missing features and the other features are filled with zeros. Considering $\mathbf{Z}^{(v)}$ in the vth view and $\mathbf{Z}^{(w)}$ in the wth view, we introduce another mapping function $g(\cdot)$, i.e.,

$$\mathbf{C}^{(v,w)}=g\left(idx(v),\mathbf{W}^{(w)}\right). \tag{9}$$

Here, $g(\cdot)$ follows the rule that all the elements of $\mathbf{C}^{(v,w)}$ are chosen from the entries in $\mathbf{W}^{(w)}$ using the indices of the columns of the existing features, $idx(v)$, in the vth view. Hence, $\mathbf{C}^{(v,w)} \in \mathbb{R}^{n_v\times n_v}$ denotes the alignment sparse representation of the wth view corresponding to the vth view. The size of $\mathbf{C}^{(v,w)}$ is the same as that of $\mathbf{Z}^{(v)}$. Considering $\mathbf{Z}^{(w)}$, consistent information is pursued using consensus regularization item, which integrates the sparse coefficient matrix of the view and its alignment sparse representations of other views, i.e.,

$$\left\|\mathbf{Z}^{(v)}-\lambda^{(v)}\sum_{w=1,w\neq v}^{N}\mathbf{C}^{(v,w)}\right\|_F^2 \tag{10}$$

where $\lambda^{(v)}$ is a parameter of the vth view.

A discriminative dictionary plays a crucial role in sparse representation learning when we take advantage of the self-expressiveness property of high-dimensional data. Hence, we introduce discriminative dictionary learning into the sparse representation model, which is able to learn a discriminative dictionary $\mathbf{D}^{(v)} \in \mathbb{R}^{d_v\times n_v}$ from the original samples instead of using all of them as a dictionary. To prevent $\mathbf{D}^{(v)}$ from being arbitrarily large, we consider a constraint $\|\mathbf{d}_i^{(v)}\|^2 \leq 1$,

where $\mathbf{d}_i^{(v)}$ is the ith column of $\mathbf{D}^{(v)}$. Finally, we present a discriminative sparse representation model to pursue the sparse representations of the existing features across different views

$$\begin{aligned}\min_{\mathbf{D}^{(v)},\mathbf{Z}^{(v)}} \ & \frac{1}{2}\sum_{v=1}^{N}\left\|\mathbf{X}_c^{(v)}-\mathbf{D}^{(v)}\mathbf{Z}^{(v)}\right\|_F^2 \\ & +\alpha\sum_{v=1}^{N}\left\|\mathbf{Z}^{(v)}\right\|_0+\frac{\beta}{2}\left\|\mathbf{Z}^{(v)}-\lambda^{(v)}\sum_{w=1,w\neq v}^{N}\mathbf{C}^{(v,w)}\right\|_F^2 \\ \text{s.t. } & \operatorname{diag}\left(\mathbf{Z}^{(v)}\right)=0, \quad \left\|\mathbf{d}_i^{(v)}\right\|^2\leq 1,\ i=1,\ldots,n_v \end{aligned} \tag{11}$$

where β is a trade-off parameter.

B. Optimization Strategy

The discriminative dictionary $\mathbf{D}^{(v)}$ and the sparse coefficient matrix $\mathbf{Z}^{(v)}$ are learned based on minimizing problem (11), which can be effectively solved using an ADMM framework. In problem (11), all the variables associated with each view can be updated independently. Therefore, we consider the following optimization problem for the vth view:

$$\begin{aligned}\min_{\mathbf{D}^{(v)},\mathbf{Z}^{(v)}} \ & \frac{1}{2}\left\|\mathbf{X}_c^{(v)}-\mathbf{D}^{(v)}\mathbf{Z}^{(v)}\right\|_F^2 \\ & +\alpha\left\|\mathbf{Z}^{(v)}\right\|_0+\frac{\beta}{2}\left\|\mathbf{Z}^{(v)}-\lambda^{(v)}\sum_{w=1,w\neq v}^{N}\mathbf{C}^{(v,w)}\right\|_F^2 \\ \text{s.t. } & \operatorname{diag}\left(\mathbf{Z}^{(v)}\right)=0, \quad \left\|\mathbf{d}_i^{(v)}\right\|^2\leq 1,\ i=1,\ldots,n_v. \end{aligned} \tag{12}$$

We introduce an auxiliary variable $\mathbf{J}^{(v)}\in\mathbb{R}^{d_v\times n_v}$ into problem (12) and consider the following equivalent problem:

$$\begin{aligned}\min_{\mathbf{D}^{(v)},\mathbf{Z}^{(v)},\mathbf{J}^{(v)}} \ & \frac{1}{2}\left\|\mathbf{X}_c^{(v)}-\mathbf{D}^{(v)}\mathbf{J}^{(v)}\right\|_F^2 \\ & +\alpha\left\|\mathbf{Z}^{(v)}\right\|_0+\frac{\beta}{2}\left\|\mathbf{Z}^{(v)}-\lambda^{(v)}\sum_{w=1,w\neq v}^{N}\mathbf{C}^{(v,w)}\right\|_F^2 \\ \text{s.t. } & \mathbf{J}^{(v)}=\mathbf{Z}^{(v)}-\operatorname{diag}\left(\mathbf{Z}^{(v)}\right) \\ & \left\|\mathbf{d}_i^{(v)}\right\|^2\leq 1,\ i=1,\ldots,n_v \end{aligned} \tag{13}$$

where $\lambda^{(v)}$ is computed as

$$\lambda^{(v)}=\frac{1}{2\sqrt{\left\|\mathbf{J}^{(v)}-\mathbf{Z}^{(v)}\right\|_F^2}}. \tag{14}$$

The augmented Lagrangian function of problem (13) can be written as follows:

$$\begin{aligned}&\mathcal{L}\left(\mathbf{D}^{(v)},\mathbf{Z}^{(v)},\mathbf{J}^{(v)},\mathbf{Y}^{(v)}\right) \\ &=\frac{1}{2}\left\|\mathbf{X}_c^{(v)}-\mathbf{D}^{(v)}\mathbf{J}^{(v)}\right\|_F^2 \\ &\quad+\alpha\left\|\mathbf{Z}^{(v)}\right\|_0+\frac{\beta}{2}\left\|\mathbf{J}^{(v)}-\lambda^{(v)}\sum_{w=1,w\neq v}^{N}\mathbf{C}^{(v,w)}\right\|_F^2 \\ &\quad+\operatorname{tr}\left(\mathbf{Y}^{(v)^T}\left(\mathbf{J}^{(v)}-\mathbf{Z}^{(v)}+\operatorname{diag}\left(\mathbf{Z}^{(v)}\right)\right)\right) \\ &\quad+\frac{\mu}{2}\left\|\mathbf{J}^{(v)}-\mathbf{Z}^{(v)}+\operatorname{diag}\left(\mathbf{Z}^{(v)}\right)\right\|_F^2 \end{aligned} \tag{15}$$

where $\mu>0$ is an adaptive penalty parameter, and $\mathbf{Y}^{(v)}\in\mathbb{R}^{n_v\times n_v}$ is the Lagrange multiplier. The augmented Lagrangian function of problem (15) can be transformed into the following equivalent function using linear algebra techniques:

$$\begin{aligned}&\mathcal{L}\left(\mathbf{D}^{(v)},\mathbf{Z}^{(v)},\mathbf{J}^{(v)},\mathbf{Y}^{(v)}\right) \\ &=\frac{1}{2}\left\|\mathbf{X}_c^{(v)}-\mathbf{D}^{(v)}\mathbf{J}^{(v)}\right\|_F^2 \\ &\quad+\alpha\left\|\mathbf{Z}^{(v)}\right\|_0+\frac{\beta}{2}\left\|\mathbf{J}^{(v)}-\lambda^{(v)}\sum_{w=1,w\neq v}^{N}\mathbf{C}^{(v,w)}\right\|_F^2 \\ &\quad+\frac{\mu}{2}\left\|\mathbf{Z}^{(v)}-\operatorname{diag}\left(\mathbf{Z}^{(v)}\right)-\left(\mathbf{J}^{(v)}+\frac{\mathbf{Y}^{(v)}}{\mu}\right)\right\|_F^2. \end{aligned} \tag{16}$$

According to the ADMM framework, we need to update each of $\mathbf{D}^{(v)}$, $\mathbf{Z}^{(v)}$, $\mathbf{J}^{(v)}$ and $\mathbf{Y}^{(v)}$ alternately while keeping the other variables fixed until the algorithm converges. For clarity, we denote $\mathbf{D}_{(k+1)}^{(v)}$, $\mathbf{Z}_{(k+1)}^{(v)}$, and $\mathbf{J}_{(k+1)}^{(v)}$ as the optimization variables at the $(k+1)$th iteration, and $\mathbf{Y}_{(k+1)}^{(v)}$ as the Lagrange multiplier at the $(k+1)$th iteration. The updating scheme at the $(k+1)$th iteration is formulated as follows.

Updating rule for $\mathbf{J}^{(v)}$ at the $(k+1)$th iteration by setting the partial derivative of $\mathcal{L}$ in function (16) with respect to $\mathbf{J}^{(v)}$ to zero, we can obtain

$$\begin{aligned}\mathbf{J}_{k+1}^{(v)} &= \mathbf{A}\left(\mathbf{D}_k^{(v)^T}\mathbf{X}_c^{(v)}+\mu\mathbf{Z}_k^{(v)}+\beta\mathbf{B}-\mathbf{Y}_k^{(v)}\right) \\ \mathbf{J}_{k+1}^{(v)} &= \operatorname{normalize}_{(0,1]}\left(\mathbf{J}_{k+1}^{(v)}\right)\end{aligned} \tag{17}$$

where $\mathbf{A}=(\mathbf{D}_k^{(v)^T}\mathbf{D}_k^{(v)}+(\mu+\beta)\cdot\mathbf{I})^{-1}$ and $\mathbf{B}=\lambda^{(v)}\sum_{w=1,w\neq v}^{N}\mathbf{C}_k^{(v,w)}$.

Update the rule for $\mathbf{Z}^{(v)}$ at the $(k+1)$th iteration

$$\begin{aligned}\mathbf{Z}_{k+1}^{(v)} &= \min_{\mathbf{Z}_{k+1}^{(v)}}\frac{\alpha}{\mu}\left\|\mathbf{Z}_{k+1}^{(v)}\right\|_0+\frac{1}{2}\left\|\mathbf{Z}_{k+1}^{(v)}-\left(\mathbf{J}_{k+1}^{(v)}+\frac{\mathbf{Y}_k^{(v)}}{\mu}\right)\right\|_F^2 \\ \mathbf{Z}_{k+1}^{(v)} &\leftarrow \mathbf{Z}_{k+1}^{(v)}-\operatorname{diag}\left(\mathbf{Z}_{k+1}^{(v)}\right).\end{aligned} \tag{18}$$

The hard thresholding operator $\mathcal{T}_{\sqrt{\eta}}(x)$ applied to each element of a given matrix is defined as follows [1]:

$$\mathcal{T}_{\sqrt{\eta}}(x)=\begin{cases}0, & \text{if } |x|\leq\sqrt{\eta}\\ x, & \text{if } |x|>\sqrt{\eta}.\end{cases} \tag{19}$$

The closed-form solution of the first problem in (18) is obtained using the operator $\mathcal{T}$

$$\mathbf{Z}_{k+1}^{(v)}=\mathcal{T}_{\sqrt{\frac{\alpha}{\mu_k}}}\left(\mathbf{J}_{k+1}^{(v)}+\frac{\mathbf{Y}_k^{(v)}}{\mu_k}\right). \tag{20}$$

Updating rule for $\mathbf{D}^{(v)}$ at the $(t+1)$th iteration, the updated dictionary $\mathbf{D}_{k+1}^{(v)}$ is accordingly expressed as

$$\mathbf{D}_{k+1}^{(v)}=\min_{\mathbf{D}_{k+1}^{(v)}}\frac{1}{2}\left\|\mathbf{X}_c^{(v)}-\mathbf{D}_{k+1}^{(v)}\mathbf{J}_{k+1}^{(v)}\right\|_F^2. \tag{21}$$

According to the constraint $\mathbf{J}^{(v)}=\mathbf{Z}^{(v)}-\operatorname{diag}(\mathbf{Z}^{(v)})$ in problem (13), we consider $\mathbf{Z}_{k+1}^{(v)}$ in the second expression of (18) to be a surrogate for $\mathbf{J}_{k+1}^{(v)}$ in problem (18), which is beneficial to

achieve the dictionary $\mathbf{D}_{k+1}^{(v)}$ for the effective reconstruction of the features in the sparse representation. Hence, we have

$$\psi\left(\mathbf{D}_{k+1}^{(v)}\right) = \frac{1}{2}\left\|\mathbf{X}_c^{(v)} - \mathbf{D}_{k+1}^{(v)}\mathbf{Z}_{k+1}^{(v)}\right\|_F^2. \tag{22}$$

The update rule of the first-order stochastic gradient descent for $\mathbf{D}_{k+1}^{(v)}$ is

$$\mathbf{D}_{k+1}^{(v)} \leftarrow \mathbf{D}_{k+1}^{(v)} - \frac{1}{\tau}\cdot\frac{\partial\psi\left(\mathbf{D}_{k+1}^{(v)}\right)}{\partial\mathbf{D}_{k+1}^{(v)}} \tag{23}$$

where $(1/\tau)$ is a gradient step. In practice, each column of $\mathbf{D}_{k+1}$ is calculated individually. We set $\tau = \mathbf{A}[j, j]$ while calculating the jth column of $\mathbf{D}_{k+1}$, where $\mathbf{A} = \mathbf{Z}_k\mathbf{Z}_k^T$ [26]. The gradient of $\psi(\mathbf{D}_{k+1}^{(v)})$ with respect to $\mathbf{D}_{k+1}^{(v)}$ can be computed as follows:

$$\frac{\partial\psi\left(\mathbf{D}_{k+1}^{(v)}\right)}{\partial\mathbf{D}_{k+1}^{(v)}} = \mathbf{D}_{k+1}^{(v)}\mathbf{Z}_{k+1}^{(v)}{\mathbf{Z}_{k+1}^{(v)}}^T - \mathbf{X}_{k+1}^{(v)}{\mathbf{Z}_{k+1}^{(v)}}^T. \tag{24}$$

For constraint $t\|\mathbf{d}_i^{(v)}\|^2 \le 1$ in problem (13), we introduce the following constrained optimization:

$$\mathbf{D}^{(v)}(:, i) \leftarrow \frac{\mathbf{D}^{(v)}(:, i)}{\max\left\{1, \left\|\mathbf{D}^{(v)}(:, i)\right\|_2^2\right\}}. \tag{25}$$

Updating the rule for $\mathbf{Y}^{(v)}$ and μ_{k+1} at the $(k+1)$th iteration, given $\mathbf{Z}_{k+1}^{(v)}$, $\mathbf{J}_{k+1}^{(v)}$ and μ_k, the Lagrange multiplier $\mathbf{Y}_{k+1}^{(v)}$ is updated to

$$\mathbf{Y}_{k+1}^{(v)} = \mathbf{Y}_k{}^{(v)} + \mu_k\left(\mathbf{Z}_{k+1}^{(v)} - \mathbf{J}_{k+1}^{(v)}\right). \tag{26}$$

The penalty parameter μ_{k+1} is updated based on

$$\mu_{k+1} = \min(\rho\mu_k, \mu_{\max}) \tag{27}$$

where ρ, μ_1, and $\mu_{\max}$ are constants.

These steps are repeated until the convergence condition is satisfied for each view, i.e., $\|\mathbf{Z}_{k+1}^{(v)} - \mathbf{J}_{k+1}^{(v)}\|_{\max} < \varepsilon$, or the number of iterations exceeds a maximum iteration number. According to the mapping function $h(\cdot)$, we finally obtain $\mathbf{W}^{(v)}$ using $\mathbf{Z}_{k+1}^{(v)}$ and $\mathbf{M}^{(v)}$. Algorithm 2 summarizes the entire optimization procedure for problem (11).

Algorithm 2 Solving (11) Using an ADMM Framework

Input: Data matrices $\mathbf{X} = \left\{\mathbf{X}^{(v)}\right\}_{v=1}^N$ and $\mathbf{M} = \left\{\mathbf{M}^{(v)}\right\}_{v=1}^N$, parameters $\alpha > 0$ and $\beta > 0$.
Initialize: $\rho = 1.1$, $\mu_1 = 10^{-2}$, $\mu_{\max} = 10^6$, $\varepsilon = 10^{-6}$, $k = 1$ and $maxIters = 500$.

```
for v = 1 to N do
    X_c^(v) consists of the non-zero columns of X^(v)M^(v);
    D_1^(v) = X_c^(v);
    Z_1^(v) = J_1^(v) = Y_1^(v) = 0;
end for
while not converged do
    update J_{k+1}^(v) using (17);
    update Z_{k+1}^(v) using (18);
    update D_{k+1}^(v) using (23), (24) and (25);
    update the multiplier using (26);
    update the parameter using (27);
    /* check the convergence condition for each view: */
    converged = true;
    for v = 1 to N do
        if ||Z_{k+1}^(v) - J_{k+1}^(v)||_max >= ε then
            converged = false;
        end if
    end for
    if k >= maxIters or converged then
        break;
    end if
    k ← k + 1;
end while
for v = 1 to N do
    W^(v) = h(Z_{k+1}^(v), M^(v))
end for
```

Output: $\mathbf{W}^{(v)}$.

C. Sparsity Augmented Fusion of Sparse Representations

As analyzed above, $\mathbf{W}^{(v)} \in \mathbb{R}^{n\times n}$ cannot characterize the full similarity of the features due to the missing features of the samples in the vth view. However, the features of these samples are still preserved in the other views. This indicates that complementary information of multiple views is contained in the sparse representations of different views. With sparse representations of the incomplete multiview data, we manually perform sparsity augmented fusion of sparse representations without tuning parameters. Suppose $\mathbf{W} \in \mathbb{R}^{n\times n}$ is a sparsity augmented similarity matrix across multiple views. We first present a simple but effective sparse representation fusion scheme for IMVC. We achieve the fused sparse representation result by averaging the nonzero elements of the sparse representations in multiple views. Specifically, each of the elements in $\mathbf{W}$ can be calculated by

$$\mathbf{W}_{ij} = \frac{1}{n_{ij}}\sum_{v=1}^{N}\mathbf{W}_{ij}^{(v)} \tag{28}$$

where n_{ij} is the number of nonzero elements in $\mathbf{W}_{ij}^{(v)}$.

The sparsity ratio (SR) of $\mathbf{W}$ is defined as

$$\mathrm{SR}(\mathbf{W}) = \frac{\|\mathbf{W}\|_0}{\mathrm{size}(\mathbf{W})} \tag{29}$$

where size($\mathbf{W}$) is the total number of elements in $\mathbf{W}$. From the perspective of graph connectivity, the fused result of the sparse representations in multiple views remains sparse, which ensures high correlations for samples with the same cluster. Unfortunately, the positions of the missing features are often inconsistent in different views. The sparsity of $\mathbf{W}$ cannot be guaranteed even though $\mathbf{W}^{(v)}$ is sparse in each view. Moreover, the sparsity of the result of Algorithm 2 is determined using (20). The parameter α is not known, and μ_k always increases during the iterations of Algorithm 2. The sparsity of $\mathbf{W}^{(v)}$ may not be achieved in Algorithm 2. Consequently, we take into

account the sparsity in the fusion of the sparse representations across different views.

We propose an alternative fusion scheme, namely, sparsity augmented fusion of sparse representations, when SR($\mathbf{W}$) is greater than a given value. The sparsity augmented similarity matrix $\mathbf{W}$ can be achieved by solving the following problem:

$$\min_{\mathbf{W}} \ \tau\|\mathbf{W}\|_1 + \left\|\mathbf{W} - \sum_{v=1}^{N}\theta^{(v)}\mathbf{W}^{(v)}\right\|_F^2 \quad \text{s.t. } \mathbf{w}_{ij} \geq 0, \quad \mathbf{1}^T\mathbf{w}_i = 1 \tag{30}$$

where τ is a parameter, $\theta^{(v)}$ is a positive scalar, $\mathbf{w}_i \in \mathbb{R}^n$ represents the ith column of $\mathbf{W}$, and w_{ij} is the jth element of $\mathbf{w}_i$. This scheme adopts the weight of the vth view $\theta^{(v)}$ to consider the individual importance of each view in the sparse representation fusion of multiple views.

The first term $\|\mathbf{W}\|_1$ is a constant in problem (30) according to the definition of $\|\cdot\|_1$ and the constraint $\mathbf{1}^T\mathbf{w}_i = 1$. Hence, the optimization of problem (30) is simplified into

$$\min_{\mathbf{W}} \left\|\mathbf{W} - \sum_{v=1}^{N}\theta^{(v)}\mathbf{W}^{(v)}\right\|_F^2 \quad \text{s.t. } \mathbf{w}_{ij} \geq 0, \quad \mathbf{1}^T\mathbf{w}_i = 1. \tag{31}$$

When $\theta^{(v)}$ is given, the optimization of each vector $\mathbf{w}_i$ that represents the ith column of $\mathbf{W}$ is independent in problem (31). We formulate problem (31) in a vector form as follows:

$$\min_{\mathbf{w}_i} \left\|\mathbf{w}_i - \sum_{v=1}^{N}\theta^{(v)}\mathbf{w}_i^{(v)}\right\|_2^2 \quad \text{s.t. } \mathbf{w}_{ij} \geq 0, \quad \mathbf{1}^T\mathbf{w}_i = 1. \tag{32}$$

The optimal $\mathbf{w}_i$ has a closed-form solution, which can be efficiently obtained with the guarantee of sparsity using Algorithm 1 [11].

In problem (31), $\theta^{(v)}$ is considered to be a positive weight to balance the significance of the vth view. Here, we further exploit the adaptive weight of the vth view $\theta^{(v)}$ as follows:

$$\min_{\mathbf{w}_{ij}} \ \sum_{v=1}^{N}\theta^{(v)}\sum_{i,j=1}^{n}\left(\mathbf{w}_{ij} - \mathbf{w}_{ij}^{(v)}\right)^2 \quad \text{s.t. } \mathbf{w}_{ij} \geq 0, \quad \mathbf{1}^T\mathbf{w}_i = 1. \tag{33}$$

The Lagrangian function of problem (33) is written as

$$\mathcal{L}(\mathbf{w}_{ij}, \gamma) = \sum_{v=1}^{N}\theta^{(v)}\sum_{i,j=1}^{n}\left(\mathbf{w}_{ij} - \mathbf{w}_{ij}^{(v)}\right)^2 - \gamma\left(\mathbf{1}^T\mathbf{w}_i - 1\right) - \mathbf{p}^T\mathbf{w}_i \tag{34}$$

where γ is a Lagrange multiplier, and $\mathbf{p} \in \mathbb{R}_+^n$ is a nonnegative Lagrange vector. Taking the partial derivative of the Lagrangian function with respect to w_{ij} and setting the partial derivative to zero for the vth view, we have

$$\frac{\partial\mathcal{L}}{\partial\mathbf{w}_{ij}} = 2\theta^{(v)}\sum_{i,j=1}^{n}\left(\mathbf{w}_{ij} - \mathbf{w}_{ij}^{(v)}\right) - \gamma - \mathbf{p}_j = 0. \tag{35}$$

According to the complementary slackness Karush–Kuhn–Tucker (KKT) condition, we have $\mathbf{p}_j = 0$ whenever $\mathbf{w}_{ij} > 0$. Thus, if $\gamma = 1$ we get that

$$\theta^{(v)} = \frac{1}{2\sum_{i,j=1}^{n}\left(\mathbf{w}_{ij} - \mathbf{w}_{ij}^{(v)}\right)}. \tag{36}$$

Algorithm 3 ASR Method

Input: Data matrices $\mathbf{X} = \{\mathbf{X}^{(v)}\}_{v=1}^N$ and $\mathbf{M} = \{\mathbf{M}^{(v)}\}_{v=1}^N$, the number of clusters c, parameters α and β.

Initialize: $ratio = 0.2$, $\varepsilon = 5e^{-2}$, $\mathbf{W}_1 = \mathbf{0}$, $\left\{\theta_1^{(v)}\right\}_{v=1}^N = 0$, $k = 1$ and $maxIters = 500$.

```
Solve Problem (11) using Algorithm 2 and obtain the optimal solution {W^(v)}_{v=1}^N.
Calculate W using (28);
Compute the sparsity ratio SR(W) using (29);
if SR(W) > ratio then
  while not converged do
    for v = 1 to N do
      Update θ_{k+1}^(v) using (36);
    end for
    Compute W_sum = Σ_{v=1}^N θ_{k+1}^(v) W^(v);
    Apply W_sum and σ = 1 in Algorithm 1 to obtain W_{k+1};
    Check the convergence condition using (38);
    if k > maxIters or converged then
      break;
    end if
    k ← k + 1;
  end while
end if
Apply W in the NCuts algorithm, and obtain c clusters of X;
```

Output: The c clusters.

We introduce an alternating optimization algorithm that iteratively obtains $\mathbf{W}$ by solving problem (31) and updates $\theta^{(v)}$ using (36). The two steps are independent during iterative computations. Let

$$f\left(\mathbf{W}, \theta^{(v)}\right) = \sum_{v=1}^{N}\theta^{(v)}\sum_{i,j=1}^{n}\left(\mathbf{w}_{ij} - \mathbf{w}_{ij}^{(v)}\right)^2 \tag{37}$$

be the objective value of the combination of problems (31) and (33). The convergence condition is

$$\left|f\left(\mathbf{W}_k, \theta_k^{(v)}\right) - f\left(\mathbf{W}_{k+1}, \theta_{k+1}^{(v)}\right)\right| < \varepsilon \tag{38}$$

where $\varepsilon = 5e^{-2}$. Algorithm 3 summarizes the overall procedures of the proposed method.

D. Theoretical Analysis

1) Convergence Analysis: We first discuss the convergence condition in Algorithm 2. It would not be easy to prove the convergence of Algorithm 2 in theory since the objective function of problem (11) is not smooth. Fortunately, the value of the convergence condition $\|\mathbf{Z}_{k+1}^{(v)} - \mathbf{J}_{k+1}^{(v)}\|_{\max}$ steadily declines as the number of iterations gradually increases in our experiments. Hence, the convergence condition is finally satisfied for each view under the ADMM framework, i.e., $\|\mathbf{Z}_{k+1}^{(v)} - \mathbf{J}_{k+1}^{(v)}\|_{\max} < \varepsilon$, which is demonstrated in our experiments. Then, we estimate the convergence condition in Algorithm 3. The convergence of Algorithm 3 is proven by Theorem 1.

Theorem 1: The value of (38) will monotonically decrease in each iteration until convergence.

Proof: Suppose $\mathbf{w}_i^k$ and $\mathbf{w}_i^{k+1}$ are two optimal solutions of problem (32) at k and $k+1$, respectively. Then,

$$\begin{aligned} & f\left(\mathbf{W}_k, \theta_k^{(v)}\right) - f\left(\mathbf{W}_{k+1}, \theta_{k+1}^{(v)}\right) \\ &= \theta_k^{(v)} \sum_{i,j=1}^{n} \left(\mathbf{w}_{ij}^k - \mathbf{w}_{ij}^{(v)}\right)^2 - \theta_{k+1}^{(v)} \sum_{i,j=1}^{n} \left(\mathbf{w}_{ij}^{k+1} - \mathbf{w}_{ij}^{(v)}\right)^2 \\ &= \sum_{i,j=1}^{n} \left(\mathbf{w}_{ij}^k - \mathbf{w}_{ij}^{k+1}\right). \end{aligned}$$

Considering $\varphi = (1/m)(\sum_{j=1}^{m} \mathbf{w}_{ij} - \sigma)$, $m \leq n$, and $\sigma = 1$ in Algorithm 1, we get $\varphi \leq 0$ after the first iteration.

According to the output of Algorithm 1, we have

$$\begin{aligned} \mathbf{w}_{ij}^k - \mathbf{w}_{ij}^{k+1} &= \mathbf{w}_{ij}^k - \max\left(\mathbf{w}_{ij}^k - \varphi_{k+1}, 0\right) \\ &= \varphi_{k+1} \leq 0. \end{aligned}$$

Assume that the claim $\varphi_{k+1} < 0$ holds. According to the constraints $\mathbf{w}_{ij} \geq 0$, we will have

$$\sum_{j=1}^{n} \mathbf{w}_{ij}^k > 1$$

if k increases steadily. However, this does not hold because of contradicting $\mathbf{1}^T \mathbf{w}_i = 1$. This indicates that the following equality holds:

$$\left|\mathbf{w}_{ij}^k - \mathbf{w}_{ij}^{k+1}\right| < \varepsilon$$

as k gradually increases, where ε is a small positive constant. Hence, convergence will eventually be achieved as the value of (38) monotonically decreases. □

2) Complexity Analysis: In Algorithm 2, three important variables $\mathbf{J}^{(v)}$, $\mathbf{Z}^{(v)}$, and $\mathbf{D}^{(v)}$ are updated alternately during each iteration. The first step of Algorithm 2 that updates $\mathbf{J}^{(v)}$ requires the inverse operation of an $n \times n$ matrix, whose computational complexity is $\mathcal{O}(n^3)$. The second and third steps that update $\mathbf{Z}^{(v)}$ and $\mathbf{D}^{(v)}$ take a computational complexities of $\mathcal{O}(n^2)$ and $\mathcal{O}(\sum_{v=1}^{N} d_v n^3)$ in Algorithm 2, respectively. The overall computational complexity of Algorithm 2 is $\mathcal{O}(N(1 + \sum_{v=1}^{N} d_v)n^3)$ for each iteration. In addition, there are two critical steps of iteratively updating $\theta^{(v)}$ and $\mathbf{W}$ in Algorithm 3, which require computational complexities of $\mathcal{O}(Nn^2)$ and $\mathcal{O}(n^2 \log(n))$, respectively. Therefore, the final overall complexity of Algorithm 3 is $\mathcal{O}((t_1 N(1 + \sum_{v=1}^{N} d_v)n^3) + t_2((N + \log(n))n^2))$, where t_1 and t_2 are the numbers of iterations.

3) Sparse Representation Fusion Analysis: We analyze how to ensure the sparsity of the fused result generated from the sparse representation fusion in Algorithm 3. The solution of problem (31) is equivalent to that of problem (30). The sparsity of the solution of problem (31) is guaranteed by Algorithm 1. In addition, we further discuss the sparsity stability of the solution of problem (31) during the iterations in Algorithm 3. In Algorithm 1, we sort $\mathbf{z}^t$ into $\mathbf{v}^t$: $\mathbf{v}_1 \geq \mathbf{v}_2 \geq \cdots \geq \mathbf{v}_{m_k} \geq 0$. According to Theorem 1, $\mathbf{w}_{ij}^k$ steadily increases as k increases before convergence. As a result, m_k gradually declines before convergence in Algorithm 3. This shows that the sparsity stability of the fused result can be guaranteed during the iterations.

4) Extension to MVC: We present an extension to the proposed ASR method for complete multiview data. Specifically, the diagonal indicator matrix $\mathbf{M}^{(v)}$ becomes an identity matrix of size $n \times n$ in (5) if all the features are available in multiple views. Similarly, the two mapping functions $h(\cdot)$ and $g(\cdot)$ degenerate into the equivalent mapping in (8) and (9), respectively. In essence, the extension is a special case of the ASR method for MVC. This enables the proposed method to simultaneously exploit the intrinsic structures of complete or incomplete multiview data.

TABLE I
STATISTICS OF THE DATASETS

Dataset	Clusters	Views	Data samples
BBC	5	4	685
Reuters	6	5	600
Flower17	17	7	1,360
ProteinFold	27	12	694
100leaves	100	3	1,600
Caltech101	101	6	8,677

IV. EXPERIMENTS

In this section, we evaluate our ASR method on publicly available datasets for IMVC. The proposed ASR method is implemented in the MATLAB 2019b. The MATLAB source code for our method is available online.[1] We conduct experiments on a Windows 10 platform with an Intel i7-10700 CPU, 32 GB of RAM and MATLAB runtime environment.

A. Experiment Settings

1) Datasets: We use six multiview benchmark datasets to evaluate the proposed method. The important statistics of the datasets are summarized in Table I. The descriptions of the datasets are given as follows.

1) *BBC Dataset [14]:* It is collected from the BBC news website and consists of 685 documents. This dataset includes four views and was manually annotated with one of the five topical labels.
2) *Reuters Dataset [35]:* It consists of 600 documents written in five languages and their translations over a common set of six categories. We randomly select 600 documents for this dataset, with each class containing 100 documents.
3) *ProteinFold Dataset*[2]*:* It includes 12 views, each of which contains 694 protein domains that belong to 27-fold classes.
4) *Flower17 Dataset [30]:* It includes 17 different flower categories with 80 images for each class. Each image is represented by seven views.
5) *100Leaves Dataset [27]:* It contains 1600 samples from 100 categories. The shape descriptor, fine scale margin, and texture histogram features constitute three views to depict the samples from different perspectives.
6) *Caltech101 Dataset [10]:* It contains 8677 images of objects that belong to 101 classes. Each object has

[1] https://codeocean.com/capsule/2092701/tree/v1
[2] http://mkl.ucsd.edu/dataset/protein-fold-prediction

approximately 30–800 images. We removed the background category.

We first consider all the features available in multiple views. Then, we randomly remove certain features from each view. The percentage of missing rates for each view is varied from 10% to 50% in intervals of 20%. For each sample, at least one feature is preserved in one of multiple views. We applied a principal component analysis (PCA) algorithm to preprocess the original features of the samples and then normalize them in the multiview data [36].

2) Comparison Methods: We compare the proposed ASR method with several state-of-the-art IMVC methods, including the doubly aligned IMVC (DAIMC) algorithm [15], self-representation subspace clustering (SRSC) algorithm [22], generalized IMVC (GIMC) algorithm [40], efficient and effective IMVC (EE-IMVC) algorithm [23], and autoweighted noisy and IMVC (ANIMC) algorithm [13]. Moreover, we also consider an important extension of SSC for comparison, namely, $\text{SSC}_{\mathbf{Agg}}$. Specifically, we first aggregate all the sparse representation matrices, which are generated independently using SSC from each individual view, into a single matrix. Each element of the matrix represents the similarity between two corresponding samples. Similar to the proposed method, the missing features are also removed from the original data in $\text{SSC}_{\mathbf{Agg}}$. Then, we perform standard spectral clustering on the matrix. For the competing methods, the source codes were provided by their authors.

3) Evaluation Metrics: Following the previous work [5], three standard metrics were adopted to evaluate the clustering performance of all the competing algorithms, including clustering accuracy (ACC), normalized mutual information (NMI), and F-measure. In the experiments, a higher value of these metrics indicates better clustering quality.

4) Parameter Settings: There are two initial parameters α and β in the ASR method. The initial parameter α is tuned in the range of $\{1e^{-7}, 5e^{-7}, 1e^{-6}, 5e^{-6}, 1e^{-5}, 5e^{-5}, 1e^{-4}\}$, while the initial parameter β is tuned in the range of $\{1e^{-3}, 5e^{-3}, 1e^{-2}, 5e^{-2}, 0.1, 0.5, 1, 5, 10, 20\}$. Then we may slightly tune the parameters to obtain the best results. For the competing methods, the best results were obtained by manually tuning the parameters. For all the methods, we repeated each experiment ten times and reported the average results and the standard deviations. The best and second best clustering results are shown in bold and underlined, respectively.

5) Estimation of the Number of Clusters: Determining the number of clusters without any prior knowledge of data distribution remains challenging. The accurate estimation of the number of clusters can be determined if $\mathbf{W}$ is a strictly block-diagonal matrix in problem (30). The number of clusters can be obtained by counting the number of zero singular values of $\mathbf{L}$ [25], which is defined as follows:

$$\mathbf{L} = \mathbf{I} - \mathbf{Q}^{-\frac{1}{2}}\mathbf{W}\mathbf{Q}^{-\frac{1}{2}} \tag{39}$$

where $\mathbf{I} \in \mathbb{R}^{n\times n}$ is an identity matrix, and $\mathbf{Q}$ is a degree matrix for $\mathbf{W}$, i.e., $q_i = \sum_{j=1}^{n} \mathbf{W}_{ij}$. However, the structure of $\mathbf{W}$ is often not sufficiently flexible to satisfy this requirement in practice. Following the settings of the comparison methods, we assume that the number of clusters is known for each dataset.

B. Evaluation of the Clustering Performance

1) Performance Comparison: We evaluated the proposed algorithm on six multiview datasets. Table II shows the average and standard deviation values of ACC (%), NMI (%), and F-measure (%) of the ASR and the competing methods with different missing ratios on the datasets. As shown in Table II, the ASR algorithm consistently achieved the best results with respect to ACC, NMI, and F-measure when compared with the other algorithms in the experiments. For example, the ASR algorithm achieved significant improvements of at least 3.65%, 3.35%, 3.35%, and 4.09% in terms of ACC when compared with the other algorithms with different missing rates of 0%, 10%, 30%, and 50% on the BBC dataset, respectively. Moreover, we observed the same advantages of our proposed method in terms of ACC, NMI and F-measure in comparison to the other algorithms on the BBC dataset. Similarly, the ASR method outperforms the competing methods in terms of NMI and F-measure on the other four datasets. This confirms that with respect to ACC, NMI, and F-measure in the experiments, our proposed method is very effective with varying missing ratios.

As expected, the clustering performance usually declines as the missing ratio increases in the experiments. In addition, $\text{SSC}_{\mathbf{Agg}}$ often obtains comparable or even better clustering results than some other compared methods in the experiments. This demonstrates the advantage of learning sparse representation for measuring similarity among data samples. Similarly, the SRSC method, which uses the self-representation property of the high-dimensional data, achieves impressive clustering results on all the datasets, especially on the ProteinFold and 100leaves datasets. Both $\text{SSC}_{\mathbf{Agg}}$ and the SRSC method enjoy a better clustering performance over the NMF-based method, graph-learning-based method, and multiple kernel-based method. This implies that subspace learning methods are effective in measuring the similarity of data samples of incomplete multiview data in the experiments. Moreover, the proposed method performs significantly better than $\text{SSC}_{\mathbf{Agg}}$ and the SRSC method in the experiments. The differences in clustering performance between the proposed method and the competing methods become larger as the numbers of samples and views increase in the experiments. For example, the proposed method achieves better performances on the Flower17 and 100leaves datasets. This further validates that ASR effectively obtains consistent and complementary information among different views, which is beneficial to improving the clustering performance.

2) Investigating the Running Time: The running time of the proposed algorithm was also evaluated, and the results are presented in Table III. We can see that EE-IMVC performs more efficiently than the other algorithms. The running time of the proposed algorithm drops significantly as the missing ratio increases in the ASR algorithm. In addition, the ASR method achieves the second best running times with a 50% missing ratio in the Flower17, ProteinFold, and 100leaves datasets.

TABLE II

AVERAGE CLUSTERING RESULTS AND STANDARD DEVIATIONS OF THE DIFFERENT METHODS WITH VARIOUS MISSING RATIOS ON SIX MULTIVIEW DATASETS

Datasets	Methods	ACC				NMI				F-measure			
		0	0.1	0.3	0.5	0	0.1	0.3	0.5	0	0.1	0.3	0.5
BBC	SSC_{Agg}	91.97±0.00	90.22±0.00	89.64±0.00	81.75±0.00	77.96±0.00	74.75±0.00	72.15±0.00	56.15±0.00	91.99±0.00	90.2±0.00	89.61±0.00	81.48±0.00
	DAIMC	81.02±0.00	75.18±0.00	73.86±0.00	68.46±0.00	65.3±0.00	63.01±0.00	48.66±0.00	42.98±0.00	81.71±0.00	76.57±0.00	75.35±0.00	69.92±0.00
	SRSC	91.11±0.05	91.1±0.00	89.2±0.00	82.63±0.00	75.71±0.10	75.46±0.00	70.81±0.00	56.73±0.00	91.11±0.05	91.07±0.00	89.19±0.00	82.74±0.00
	GIMC	83.94±0.00	82.04±0.00	82.92±0.00	77.23±0.00	67.7±0.00	63.27±0.00	64.89±0.00	58.56±0.00	83.53±0.00	81.86±0.00	82.71±0.00	77.78±0.00
	EE-IMVC	70.02±4.74	70.89±4.96	68.89±4.37	58.32±2.91	54.07±2.71	53.97±2.91	49.97±2.60	39.54±1.66	72.7±2.69	72.87±2.88	70.61±2.68	64.14±0.78
	ANIMC	72.85±2.99	50.5±4.22	39.74±3.49	34.41±2.83	51.42±3.21	28.01±3.54	15.83±2.23	11.1±2.76	73.15±2.87	55.14±3.07	45.91±1.62	37.22±3.47
	ASR	**95.62±0.00**	**94.45±0.00**	**92.99±0.00**	**86.72±0.00**	**86.84±0.00**	**83.27±0.00**	**80.93±0.00**	**69.14±0.00**	**95.64±0.00**	**94.46±0.00**	**93.1±0.00**	**87.48±0.00**
Reuters	SSC_{Agg}	54.73±0.58	56.17±0.00	55.23±0.40	40.82±0.20	34.79±0.34	36.05±0.00	38±0.04	26.41±0.64	54.8±0.53	57.39±0.00	57.37±0.13	42.83±0.23
	DAIMC	49.78±1.86	49.25±3.72	50.05±4.55	45.88±5.41	32.15±2.8	31.91±4.01	31.96±3.51	27.37±4.3	49.7±2.09	49.78±3.03	50.79±4.12	47.43±4.33
	SRSC	54.17±0.00	54±0.00	54.57±0.20	52.05±0.18	41.66±0.00	40.96±0.00	42.13±0.11	35.24±0.15	56.43±0.00	56.14±0.00	56.67±0.14	53.64±0.15
	GIMC	53.5±0.00	52.33±0.00	51.5±0.00	52±0.00	39.78±0.00	32.45±0.00	29.94±0.00	31.25±0.00	56.37±0.00	52.94±0.00	51.25±0.00	52.66±0.00
	EE-IMVC	50.37±2.04	50.18±2.66	48.5±1.80	48.25±1.61	33.2±1.63	30.96±1.50	31.63±2.15	30.91±1.71	50.33±2.45	49.92±1.84	47.88±1.86	48.25±1.82
	ANIMC	53.35±2.61	45.67±4.68	37.58±1.97	33.83±4.07	34.29±2.01	27.01±3.77	20.38±1.98	14.53±3.48	53.9±2.83	48.02±3.84	42.06±2.17	36.67±3.74
	ASR	**64.17±0.00**	**65.68±0.05**	**63.33±0.00**	**56.67±0.00**	**47.49±0.00**	**48.54±0.17**	**53.13±0.00**	**36.09±0.00**	**63.83±0.00**	**65.56±0.05**	**65.44±0.00**	**56.71±0.00**
Flower17	SSC_{Agg}	60.96±0.07	60.65±0.13	57.12±0.08	51.39±0.30	60.73±0.06	58.9±0.07	56.84±0.08	46.18±0.2	65.11±0.05	64.63±0.09	61.37±0.11	54.35±0.36
	DAIMC	47.46±1.87	46.65±1.92	43.04±1.97	34.74±1.93	49.2±1.72	48.5±1.47	42.32±0.92	32.53±1.59	50.84±1.87	50.73±1.84	46.22±1.26	37.67±1.94
	SRSC	49.67±0.53	46.79±0.76	45.57±0.41	37.9±0.40	52.78±0.42	48.51±0.40	47.06±0.17	37.78±0.43	53.44±0.47	49.35±0.63	50.1±0.27	41.86±0.04
	GIMC	48.38±0.00	46.77±0.00	44.71±0.00	44.27±0.00	50.54±0.00	48.56±0.00	48.07±0.00	45.95±0.00	52.39±0.00	50.43±0.00	49.16±0.00	47.09±0.00
	EE-IMVC	59.2±1.92	58.63±1.29	52.67±1.64	48.63±1.06	57.33±1.20	55.12±0.91	49.89±0.84	43.29±0.62	62.64±1.93	61.19±1.18	55.03±1.46	51.12±1.16
	ANIMC	45.97±1.54	37.93±1.24	34.32±1.41	31.94±2.23	47.04±1.81	36.62±1.25	32.92±0.57	29.32±1.37	49.77±2.22	41.13±1.34	37.02±1.71	34.78±2.46
	ASR	**68.27±0.19**	**64.59±0.17**	**62.07±0.26**	**52.17±0.22**	**65.63±0.20**	**61.94±0.16**	**58.8±0.23**	**48.1±0.32**	**71.16±0.17**	**68.17±0.2**	**64.78±0.26**	**55.56±0.19**
ProteinFold	SSC_{Agg}	25.99±1.09	24.42±0.78	24.65±0.64	19.27±0.79	34.28±0.76	32.35±0.70	32.52±0.72	26.36±0.59	27.82±1.32	26.3±0.85	25.96±0.4	20.95±0.87
	DAIMC	28.26±1.43	27.51±1.88	27.28±1.5	24.58±0.92	39.72±0.47	38.94±1.7	37.16±0.97	33.38±0.65	31.47±1.12	30.45±1.86	30.33±1.58	27.27±0.866
	SRSC	37.06±1.44	34.04±1.54	34.58±1.43	31.38±0.70	45.97±0.79	44.71±0.69	43.12±0.75	39.13±0.80	39.75±1.30	37.5±1.33	37.01±1.44	33.52±0.83
	GIMC	27.52±0.00	24.5±0.00	23.78±0.00	21.18±0.00	37.51±0.00	33.6±0.00	30.88±0.00	29.62±0.00	30.48±0.00	27.4±0.00	27.26±0.00	23.87±0.00
	EE-IMVC	33.07±1.9	34.4±2.20	32.13±2.22	29.71±1.82	43.25±0.78	43.63±1.13	41.32±1.56	37.9±0.71	35.68±1.87	36.69±1.97	34.53±2.37	31.54±1.74
	ANIMC	16.76±0.79	14.77±0.57	13.62±0.55	12.85±0.43	24.9±1.35	20.54±0.61	19.37±1.02	17.67±0.5	18.6±0.89	16.3±0.61	14.78±0.65	13.49±0.5
	ASR	**41.43±0.57**	**41.25±1.19**	**36.73±1.05**	**34.64±1.16**	**50.26±0.57**	**50.03±0.48**	**46.69±0.98**	**42.82±0.81**	**43.33±0.56**	**43.59±1.04**	**39.89±1.09**	**37.51±1.17**
100leaves	SSC_{Agg}	80.66±1.19	70.69±1.08	54.15±1.62	33.33±0.72	90.56±0.42	84.71±0.36	74.15±0.56	60.69±0.35	82.02±0.98	73.18±0.87	57.77±1.27	37.8±0.78
	DAIMC	77.85±2.02	60.21±1.66	45.69±2.13	31.91±1.16	90.95±1.08	80.7±1.02	69.67±1.14	58.17±0.55	80.79±1.87	64.76±1.64	50.6±1.82	36.96±1.11
	SRSC	81.02±1.16	70.71±1.14	51.16±1.48	33.88±0.85	91.94±0.68	84.82±0.41	72.66±0.59	60.93±0.56	83.57±1.14	74.58±1.00	56.33±0.87	38.98±0.78
	GIMC	79.63±0.00	66.25±0.00	46.31±0.00	24.44±0.00	91.79±0.00	84.92±0.00	71.43±0.00	53.54±0.00	82.64±0.00	69.55±0.00	49.92±0.00	27.41±0.00
	EE-IMVC	74.76±1.03	64.05±1.65	43.68±1.29	31.12±1.05	88.25±0.67	80.6±0.71	67.9±0.62	59.47±0.57	77.03±0.95	66.92±1.56	47.98±1.21	35.45±0.98
	ANIMC	65.425±1.79	26.35±0.86	21.9±0.75	17.9±0.80	83.41±0.81	54.46±0.570	48.6±0.71	43.02±0.51	08.48±1.42	30.57±0.73	26.33±0.83	23.35±0.75
	ASR	**86.41±1.09**	**78.04±0.73**	**56.31±0.81**	**40.52±0.32**	**95.14±0.49**	**89.38±0.42**	**74.34±0.3**	**62.65±0.44**	**88.83±0.92**	**80.99±0.64**	**60.94±0.69**	**46.84±0.47**
Caltech-101	SSC_{Agg}	25.98±0.51	23.51±0.51	20.63±0.49	18.25±0.37	47.28±0.12	44.63±0.20	41.26±0.19	34.22±0.17	30.76±0.48	28.3±0.67	24.53±0.54	20.93±0.60
	DAIMC	26.1±0.68	25.46±0.87	21.15±1.36	18.7±1.05	51.55±0.29	50.57±0.34	43.17±0.7	36.37±0.88	32.85±0.82	31.71±1.1	27.53±1.55	23.91±1.04
	SRSC	23.73±0.53	20±0.4	21.9 ±0.53	18.27±0.55	48.47±0.38	42.92±0.21	42.26±0.21	37.01±0.41	29.99±0.57	26.39±0.61	27.65±0.52	23.01±0.69
	GIMC	18.11±0.00	16.09±0.00	15.02±0.00	13.14±0.00	39.05±0.00	36.02±0.00	35.1±0.00	30.52±0.00	23±0.00	19.61±0.00	18.24±0.00	16.18±0.00
	EE-IMVC	27.12±0.88	26.57±0.57	21.36±0.92	18.9±0.55	51.67±0.33	50.16±0.34	43.56±0.19	36.76±0.34	33.21±1.11	31.61±0.61	27.41±0.87	23.4±0.57
	ANIMC	15.21±0.57	9.73±0.09	9.45±0.51	9.46±0.23	35.27±0.35	12.36±0.35	16.78±0.26	18.96±0.59	19.47±0.69	7.59±0.08	8.53±0.4	9.63±0.3
	ASR	**28.46±0.88**	**26.77±1.14**	**22.61±0.55**	**19.03±0.34**	**52.98±0.32**	**51.06±0.44**	**44.25±0.31**	**37.71±0.16**	**34.88±0.92**	**32.8±1.15**	**28.83±0.66**	**24.43±0.35**

TABLE III

COMPUTATIONAL TIME (IN S) OF THE DIFFERENT METHODS WITH VARIOUS MISSING RATIOS ON SIX MULTIVIEW DATASETS

Datasets	Ratio	SSC_{Agg}	DAIMC	SRSC	GIMC	EE-IMVC	ANIMC	ASR
BBC	0	14.78	6.89	1.85	11.33	**1.76**	28.08	10.61
	0.1	12.73	7.73	1.97	10.88	**1.62**	20.35	9.30
	0.3	7.3	5.61	2.11	8.43	**1.98**	15.62	6.36
	0.5	3.12	3.65	2.28	5.23	**1.96**	11.37	2.99
Reuters	0	15.44	12.45	3.12	11.85	**1.55**	14.9	9.81
	0.1	10.53	12.06	3.21	9.58	**1.42**	15.37	8.44
	0.3	8.08	9.29	3.26	8.64	**1.46**	15.07	6.23
	0.5	4.45	3.4	2.97	5.29	**1.82**	8.58	3.54
Flower17	0	47.47	144.47	9.74	42.64	**7.01**	151.12	29.55
	0.1	32.45	145.27	16.84	35.43	**6.81**	172.83	25.39
	0.3	22.91	139.87	22.37	62.79	**7.3**	199.11	18.95
	0.5	16.68	134.71	31.49	52.13	**7.23**	200.28	14.07
ProteinFold	0	30.03	45.86	4.32	22.36	**3.99**	139.84	12.86
	0.1	23.98	45.26	4.73	19.72	**4.02**	223.22	10.96
	0.3	14.31	45.53	5.67	28.81	**3.84**	247.67	8.40
	0.5	7.88	46	6.39	22.42	**4.06**	243.08	6.02
100leaves	0	67.6	39.88	**20.85**	67.7	22.19	141.94	64.63
	0.1	57.11	44.15	25.13	43.25	**22.6**	135.54	52.96
	0.3	42.17	48.33	28.07	32.79	**22.1**	141.24	35.64
	0.5	30.92	46.26	37.48	23.88	**21.7**	140.3	28.79
Caltech-101	0	7893.1	1469.2	853.18	4796.1	**332.95**	9396.9	3575.7
	0.1	5869.5	1457.3	746.31	5875.7	**322.6**	9196.2	2823
	0.3	3110.7	1475.4	636.87	4605.4	**299.98**	9684.9	1514.1
	0.5	1465.3	1521.5	544.01	2802.7	**288.83**	9521.8	815.7

This is because the missing features are moved from the original data in the ASR method. In particular, the proposed method outperforms SSC_{Agg} in running time. This validates the advantages of the computational efficiency of the proposed method in terms of the sparse self-representation property of high-dimensional data. In summary, the experimental results show that the running time of the proposed method is medium among all the competing methods.

C. Parameter Sensitivity Analysis

We conducted experiments to investigate the effect of the parameters α and β of the ASR algorithm in terms of ACC and NMI. Due to space limitations, we selected two representative datasets with different missing ratios for parameter sensitivity analysis, the BBC and ProteinFold datasets. The parameter range of α is defined as $\{1e^{-7}, 5e^{-7}, 1e^{-6}, 5e^{-6}, 1e^{-5}, 5e^{-5}, and 1e^{-4}\}$ for the two datasets. The parameter ranges of β are defined as $\{5, 10, 20, 50, 100, 200, 500\}$ and $\{1e^{-3}, 5e^{-3}, 1e^{-2}, 5e^{-2}, 0.1, 0.5, 1\}$ for the BBC and ProteinFold datasets, respectively.

Figs. 2–5 show the influences of the parameters α and β of the ASR method with respect to ACC and NMI. From Figs. 2–5, we observe that the ASR method performs well with different missing ratios over a wide range of β and a fixed range of α. This demonstrates that the ASR method can

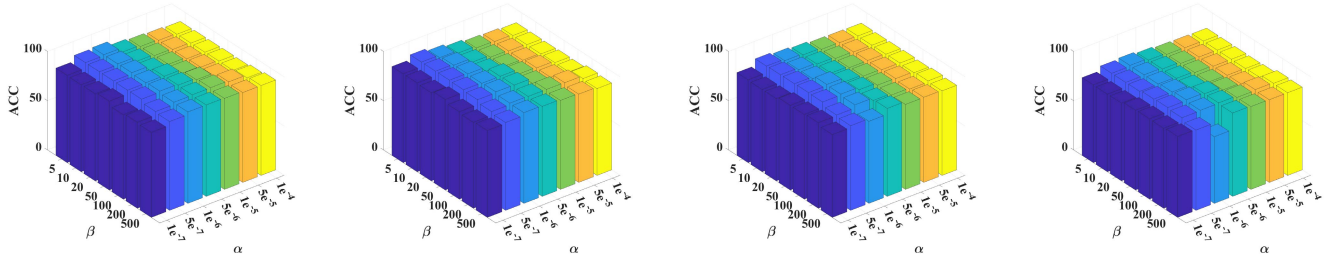

Fig. 2. ACC with different α and β combinations on the BBC dataset. (a) Ratio $= 0$. (b) Ratio $= 0.1$. (c) Ratio $= 0.3$. (d) Ratio $= 0.5$.

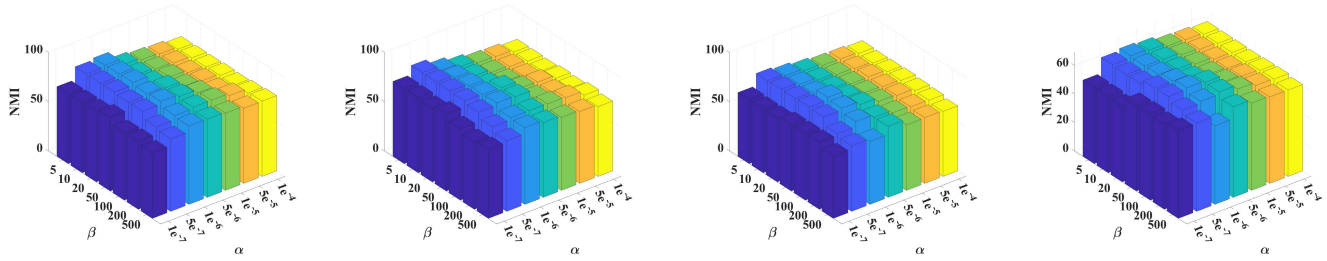

Fig. 3. NMI with different α and β combinations on the BBC dataset. (a) Ratio $= 0$. (b) Ratio $= 0.1$. (c) Ratio $= 0.3$. (d) Ratio $= 0.5$.

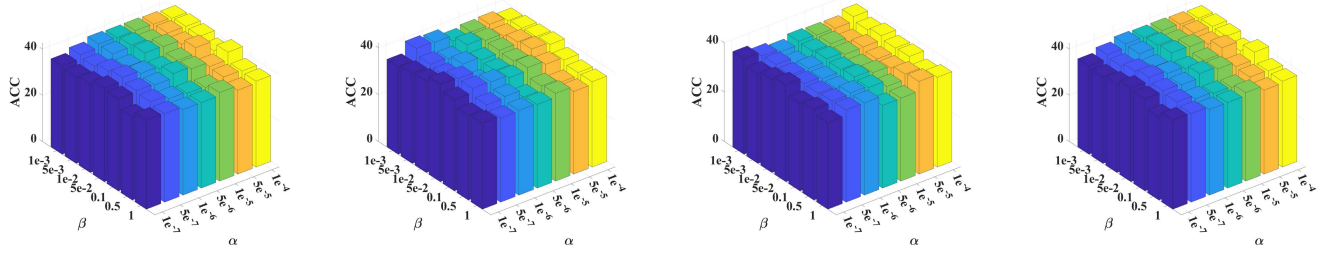

Fig. 4. ACC with different α and β combinations on the ProteinFold dataset. (a) Ratio $= 0$. (b) Ratio $= 0.1$. (c) Ratio $= 0.3$. (d) Ratio $= 0.5$.

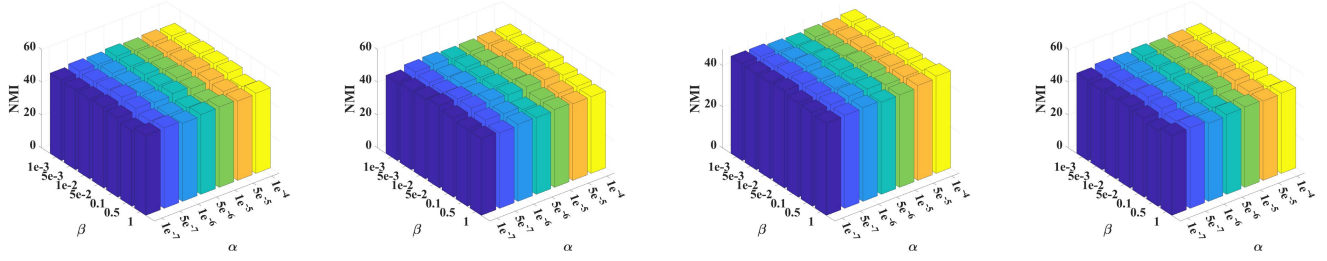

Fig. 5. NMI with different α and β combinations on the ProteinFold dataset. (a) Ratio $= 0$. (b) Ratio $= 0.1$. (c) Ratio $= 0.3$. (d) Ratio $= 0.5$.

obtain relatively stable performance across a wide range of β. However, it is slightly sensitive to the range of α due to the sparsity of results. It is still an open problem to choose a suitable range of α for different datasets without any prior knowledge. Fortunately, selecting a suitable SR for the sparsity augmented similarity matrix can help us alleviate this problem to some extent. In practice, we usually set the range of SRs to $[0.02, 0.2]$. Then, the range of α is adaptively determined according to β and SR.

D. Effectiveness of the Sparsity

We investigate the sparsity effectiveness of the fused results of sparse representations. Figs. 6 and 7 show how the SR of the sparsity augmented similarity matrix learned using sparsity augmented fusion varies with different combinations of α and β. We find that the proposed method obtains a relatively stable SR with respect to a wide range of α and β on the BBC and ProteinFold datasets. Moreover, the stability of SR of the proposed method can be demonstrated with different missing ratios. Combined with the clustering results in Figs. 2–5, the satisfied clustering results usually fall into the region in which the changes in SR fluctuate slightly. This also demonstrates that a stable SR is beneficial to the determination of α and β in practical applications.

E. Convergence Analysis

We first evaluate the stability of Algorithm 2. Specifically, the differences in the convergence among different views in

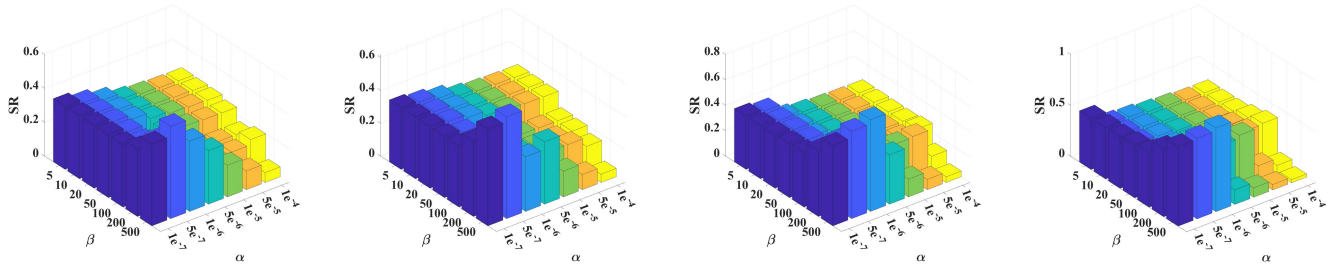

Fig. 6. SR with different α and β combinations on the BBC dataset. (a) Ratio = 0. (b) Ratio = 0.1. (c) Ratio = 0.3. (d) Ratio = 0.5.

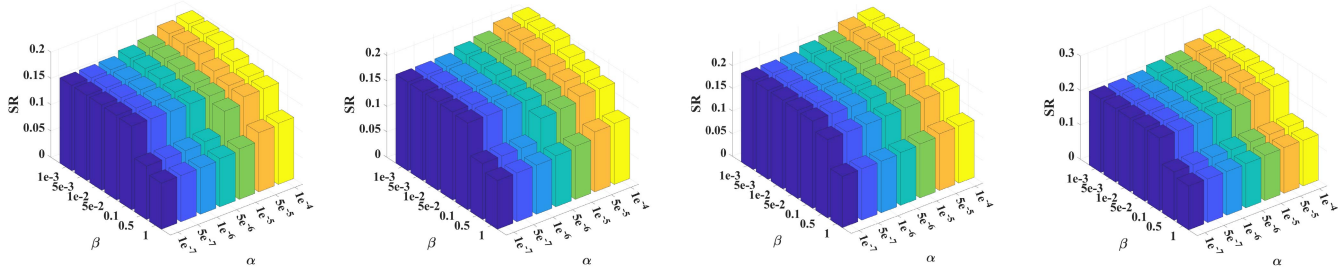

Fig. 7. SR with different α and β combinations on the ProteinFold dataset. (a) Ratio = 0. (b) Ratio = 0.1. (c) Ratio = 0.3. (d) Ratio = 0.5.

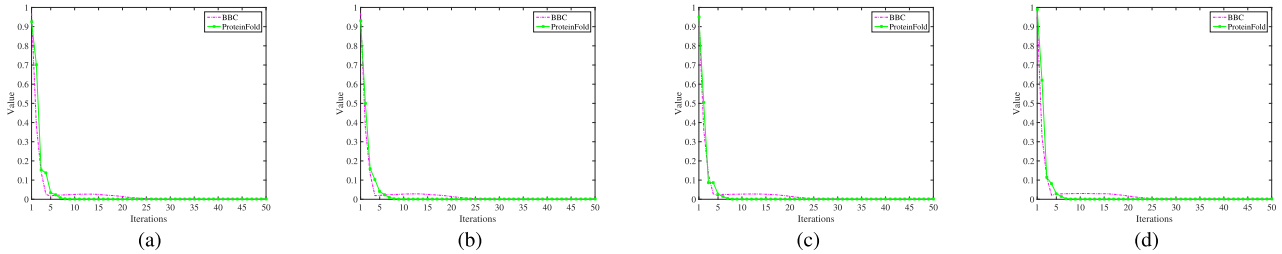

Fig. 8. Convergence results of Algorithm 2 with different missing ratios on the BBC and ProteinFold datasets. (a) Ratio = 0. (b) Ratio = 0.1. (c) Ratio = 0.3. (d) Ratio = 0.5.

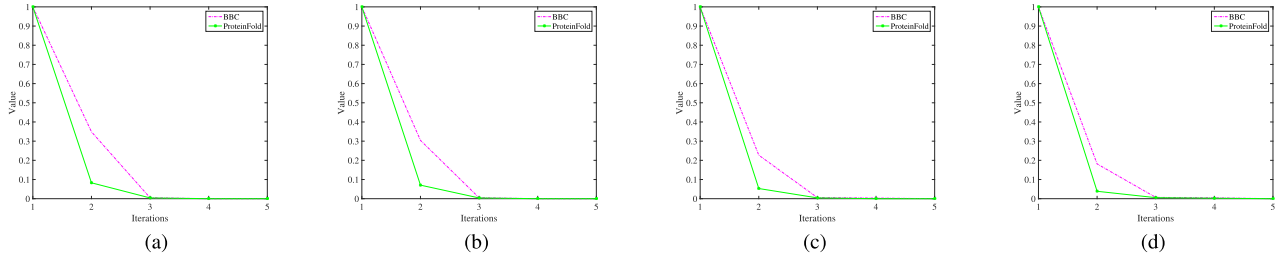

Fig. 9. Convergence results of Algorithm 3 with different missing ratios on the BBC and ProteinFold datasets. (a) Ratio = 0. (b) Ratio = 0.1. (c) Ratio = 0.3. (d) Ratio = 0.5.

Algorithm 2 are examined when the convergence condition is satisfied for each individual view. The maximum and minimum iterations of multiple views are reported in Table IV. For example, the minimum and maximum iterations of multiple views are 44 and 45 in the BBC dataset, respectively, when the missing ratio is 0. We can see that convergence can be reached in a similar number of iterations for the maximum and minimum iterations of multiple views. Thus, this indicates that Algorithm 2 has good stability during iterations.

TABLE IV

MINIMUM (MIN.) AND MAXIMUM (MAX.) ITERATIONS OF MULTIPLE VIEWS IN ALGORITHM 2

Datasets	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.
	0	0	0.1	0.1	0.3	0.3	0.5	0.5
BBC	44	45	43	45	43	45	43	44
Reuters	33	36	32	35	21	26	33	37
Flower17	16	19	16	19	16	19	16	19
ProteinFold	12	15	13	15	12	14	12	14
100leaves	71	73	67	72	56	60	59	61
Caltech-101	31	34	33	35	21	23	25	29

Then, we investigate the empirical convergence of the proposed method. Figs. 8 and 9 show the convergence curves of Algorithms 2 and 3 with different missing ratios on the BBC and ProteinFold datasets, respectively. For each figure, the x-axis represents the iteration number, and the y-axis denotes the absolute value of each convergence condition for Algorithms 2 and 3. Fig. 8 shows that Algorithm 2 usually converges quickly in dozens of iterations although the

convergence stability of Algorithm 2 is still studied in theory. Moreover, we have already proved the convergence stability of Algorithm 3 in theory. Fig. 9 shows the convergence curves of Algorithm 3 on the BBC and ProteinFold datasets. It is clear that the objective function values can quickly reach a steady state, usually within three iterations. These results demonstrate that the proposed method always converges quickly.

V. Conclusion

In this article, we present an ASR algorithm that integrates the DSRL model and a sparsity augmented fusion scheme for IMVC. In the DSRL model, the sparse regularization item and a consensus regularization item are used to explore the complementary and consistent information across multiple views, respectively. The proposed method flexibly learns the individual sparse representations of multiple views by making full use of the self-expressiveness property of the existing features in multiview data, where a discriminative dictionary is simultaneously learned to improve the compactness of sparse representations. When prior knowledge of the missing features was available, we removed them from the original data to effectively enhance the stability of the self-expressiveness property. The method is totally different from most existing IMVC methods, which either fill with zeros or the average feature values instead of the missing features. Then, a sparsity augmented similarity matrix across different views can be obtained from a sparsity augmented fusion of sparse representations, which preserves the similarity between each pair of samples. The effectiveness and importance of SR of the similarity matrix are validated in the experiments. The convergence properties of the proposed method are presented in theory and in practice. Extensive experimental results demonstrate that our method improves the IMVC performance in comparison to the state-of-the-art methods. In the future, we would like to extend the proposed method to address the clustering problem on multiview data stream clustering.

References

[1] T. Blumensath and M. E. Davies, "Iterative thresholding for sparse approximations," *J. Fourier Anal. Appl.*, vol. 14, nos. 5–6, pp. 629–654, Dec. 2008.

[2] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[3] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1548–1560, Aug. 2010.

[4] D. Cai, X. He, J. Han, and H.-J. Zhang, "Orthogonal Laplacianfaces for face recognition," *IEEE Trans. Image Process.*, vol. 15, no. 11, pp. 3608–3614, Nov. 2006.

[5] J. Chen, H. Mao, Z. Wang, and X. Zhang, "Low-rank representation with adaptive dictionary learning for subspace clustering," *Knowl.-Based Syst.*, vol. 223, pp. 1–12, Jul. 2021.

[6] J. Chen, H. Mao, H. Zhang, and Z. Yi, "Symmetric low-rank preserving projections for subspace learning," *Neurocomputing*, vol. 315, no. 13, pp. 381–393, Nov. 2018.

[7] J. Chen, S. Yang, H. Mao, and C. Fahy, "Multiview subspace clustering using low-rank representation," *IEEE Trans. Cybern.*, early access, Jun. 29, 2021, doi: 10.1109/TCYB.2021.3087114.

[8] J. Chen and Z. Yi, "Sparse representation for face recognition by discriminative low-rank matrix recovery," *J. Vis. Commun. Image Represent.*, vol. 25, no. 5, pp. 763–773, 2014.

[9] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal ℓ^1-norm solution is also the sparsest solution," *Commun. Pure Appl. Math.*, vol. 59, no. 6, pp. 797–829, Mar. 2006.

[10] D. Dua and C. Graff. (2017). UCI Machine Learning Repository. University of California Irvine. School of Information and Computer Sciences. [Online]. Available: http://archive.ics.uci.edu/ml

[11] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra, "Efficient projections onto the ℓ^1-ball for learning in high dimensions," in *Proc. 25th Int. Conf. Mach. Learn. (ICML)*, Helsinki, Finland, Jul. 2008, pp. 272–279.

[12] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, Nov. 2013.

[13] X. Fang, Y. Hu, P. Zhou, and D. O. Wu, "ANIMC: A soft approach for auto-weighted noisy and incomplete multi-view clustering," *IEEE Trans. Artif. Intell.*, vol. 3, no. 2, pp. 192–206, Aug. 2021.

[14] D. Greene and P. Cunningham, "Producing accurate interpretable clusters from high-dimensional data," in *Proc. Knowl. Discovery Databases*, Porto, Portugal, Oct. 2005, pp. 486–494.

[15] M. Hu and S. Chen, "Doubly aligned incomplete multi-view clustering," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 2262–2268.

[16] M. Hu and S. Chen, "One-pass incomplete multi-view clustering," in *Proc. AAAI Conf. Artif. Intell.*, Jan. 2019, pp. 3838–3845.

[17] Z. Huang, P. Hu, J. T. Zhou, J. Lv, and X. Peng, "Partially view-aligned clustering," in *Proc. Adv. Neural. Inf. Process. Syst.*, Dec. 2020, pp. 1–11.

[18] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient K-means clustering algorithm: Analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, Jul. 2002.

[19] A. Khan and P. Maji, "Multi-manifold optimization for multi-view subspace clustering," *IEEE Trans. Neural. Netw. Learn. Syst.*, vol. 33, no. 8, pp. 3895–3907, Feb. 2021.

[20] Z. Li, C. Tang, X. Liu, X. Zheng, W. Zhang, and E. Zhu, "Tensor-based multi-view block-diagonal structure diffusion for clustering incomplete multi-view data," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2021, pp. 1–6.

[21] Y. Lin, Y. Gou, Z. Liu, B. Li, J. Lv, and X. Peng, "Completer: Incomplete multi-view clustering via contrastive predictio," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021, pp. 11174–11183.

[22] J. Liu *et al.*, "Self-representation subspace clustering for incomplete multi-view data," in *Proc. 29th ACM Int. Conf. Multimedia*, Chengdu, China, Oct. 2021, pp. 1–11.

[23] X. Liu *et al.*, "Efficient and effective regularized incomplete multi-view clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 8, pp. 2634–2646, Aug. 2021.

[24] X. Liu *et al.*, "Late fusion incomplete multi-view clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 10, pp. 2410–2423, Oct. 2018.

[25] U. von Luxburg, "A tutorial on spectral clustering," *Statist. Comput.*, vol. 17, no. 4, pp. 395–416, Dec. 2007.

[26] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *J. Mach. Learn. Res.*, vol. 11, pp. 19–60, Mar. 2010.

[27] C. Mallah, J. Cope, and J. Orwell, "Plant leaf classification using probabilistic integration of shape, texture and margin features," *Signal Process., Pattern Recognit. Appl.*, vol. 5, no. 1, pp. 45–54, 2013.

[28] S. Matsushima and M. Brbić, "Selective sampling-based scalable sparse subspace clustering," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Vancouver, BC, Canada, Dec. 2019, pp. 12416–12425.

[29] F. Nie, H. Zhang, R. Wang, and X. Li, "Multi-view clustering: A scalable and parameter-free bipartite graph fusion method," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 44, no. 1, pp. 330–344, Jan. 2020.

[30] M.-E. Nilsback and A. Zisserman, "A visual vocabulary for flower classification," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis.*, New York, NY, USA, Jun. 2006, pp. 1447–1454.

[31] X. Peng, J. Feng, S. Xiao, W.-Y. Yau, J. T. Zhou, and S. Yang, "Structured autoencoders for subspace clustering," *IEEE Trans. Image Process.*, vol. 27, no. 10, pp. 5076–5086, Oct. 2018.

[32] X. Peng, H. Zhu, J. Feng, C. Shen, H. Zhang, and J. T. Zhou, "Deep clustering with sample-assignment invariance prior," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 11, pp. 4857–4868, Nov. 2020.

[33] J. Shi, J. Malik, and S. Sastry, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 22, no. 8, pp. 181–214, Aug. 2000.

[34] S. Shi, F. Nie, R. Wang, and X. Li, "Multi-view clustering via nonnegative and orthogonal graph reconstruction," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jul. 21, 2021, doi: 10.1109/TNNLS.2021.3093297.
[35] M. Thoma. (Jul. 2017). *The Reuters Dataset*. [Online]. Available: https://Martin-thoma.com/nlp-reuters
[36] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Maui, HI, USA, Jun. 1991, pp. 586–587.
[37] H. Wang, Y. Yang, B. Liu, and H. Fujita, "A study of graph-based system for multi-view clustering," *Knowl.-Based Syst.*, vol. 163, pp. 1009–1019, Jan. 2019.
[38] Q. Wang, Z. Ding, Z. Tao, Q. Gao, and Y. Fu, "Generative partial multi-view clustering with adaptive fusion and cycle consistency," *IEEE Trans. Image Process.*, vol. 30, pp. 1771–1783, 2021.
[39] J. Wen *et al.*, "Adaptive graph completion based incomplete multi-view clustering," *IEEE Trans. Multi.*, vol. 23, pp. 2493–2504, 2021.
[40] J. Wen, Z. Zhang, Z. Zhang, L. Fei, and M. Wang, "Generalized incomplete multiview clustering with flexible locality structure diffusion," *IEEE Trans. Cybern.*, vol. 51, no. 1, pp. 101–114, Jan. 2021.
[41] J. Wen *et al.*, "Unified tensor framework for incomplete multi-view clustering and missing-view inferring," in *Proc. AAAI Conf. Artif. Intell.*, May 2021, pp. 10273–10281.
[42] C. Xu, H. Liu, Z. Guan, X. Wu, J. Tan, and B. Ling, "Adversarial incomplete multiview subspace clustering networks," *IEEE Trans. Cybern.*, early access, Mar. 22, 2021, doi: 10.1109/TCYB.2021.3062830.
[43] Q. Yin, S. Wu, and L. Wang, "Incomplete multi-view clustering via subspace learning," in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2015, pp. 383–392.
[44] C. Zhang, Y. Cui, Z. Han, J. T. Zhou, H. Fu, and Q. Hu, "Deep partial multi-view learning," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 44, no. 5, pp. 2402–2415, Nov. 2020.
[45] C. Zhang, H. Fu, J. Wang, W. Li, X. Cao, and Q. Hu, "Tensorized multi-view subspace representation learning," *Int. J. Comput. Vis.*, vol. 128, no. 8, pp. 2344–2361, 2020.
[46] C. Zhang, Z. Han, Y. Cui, H. Fu, J. T. Zhou, and Q. Hu, "CPM-Nets: Cross partial multi-view networks," in *Proc. Adv. Neural. Inf. Process. Syst.*, Vancouver, BC, Canada, Dec. 2019, pp. 1–11.

Jie Chen (Member, IEEE) received the B.Sc. degree in software engineering and the M.Sc. and Ph.D. degrees in computer science from Sichuan University, Chengdu, China, in 2005, 2008, and 2014, respectively.

From 2008 to 2009, he was with Huawei Technologies Company, Ltd., Shenzhen, China, as a Software Engineer. He is currently an Associate Professor with the College of Computer Science, Sichuan University. His current research interests include machine learning, big data analysis, and deep neural networks.

Shengxiang Yang (Senior Member, IEEE) received the Ph.D. degree from Northeastern University, Shenyang, China, in 1999.

He is currently a Professor of computational intelligence and the Director of the Centre for Computational Intelligence, School of Computer Science and Informatics, De Montfort University, Leicester, U.K. He has over 340 publications with an H-index of 63 according to Google Scholar. His current research interests include evolutionary computation, swarm intelligence, artificial neural networks, data mining and data stream mining, and relevant real-world applications.

Dr. Yang serves as an Associate Editor/Editorial Board Member for a number of international journals, such as the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, IEEE TRANSACTIONS ON CYBERNETICS, *Information Sciences*, and *Enterprise Information Systems*.

Xi Peng (Member, IEEE) is currently a Full Professor with the College of Computer Science, Sichuan University, Chengdu, China. His current research interests include machine intelligence. He has authored more than 50 articles in this area.

Dr. Peng has served as an Associate Editor/Guest Editor for six journals, including the IEEE TRANSACTIONS ON SMC: SYSTEMS and IEEE TRANSACTIONS ON NEURAL NETWORK AND LEARNING SYSTEMS and the Area Chair/Senior Program Committee Member for conferences, such as International Joint Conference on Artificial Intelligence (IJCAI), the Association for the Advance of Artificial Intelligence (AAAI), and International Conference on Multimedia and Exposition (ICME).

Dezhong Peng (Member, IEEE) received the B.Sc. degree in applied mathematics and the M.Sc. and Ph.D. degrees in computer software and theory from the University of Electronic Science and Technology of China, Chengdu, China, in 1998, 2001, and 2006, respectively.

From 2001 to 2007, he was with the University of Electronic Science and Technology of China as an Assistant Lecturer and a Lecturer. He was a Post-Doctoral Research Fellow with the School of Engineering, Deakin University, Burwood, VIC, Australia, from 2007 to 2009. He is currently a Professor with the College of Computer Science, Sichuan University, Chengdu. His research interests include blind signal processing and neural networks.

Zhu Wang received the B.M., M.Sc., and L.L.D. degrees in civil and commercial law from the Renmin University of China, Beijing, China, in 2003, 2006, and 2009, respectively.

He is currently a Professor in law and the Director of the Institute of Rule of Law of Market Economy, Sichuan University, Chengdu, China. His research interests mainly include tort, insurance law, constitution, and big data analysis of law.