Adaptive Sample-level Graph Combination for Partial Multiview Clustering

Liu Yang, Chenyang Shen, Qinghua Hu*, Liping Jing, Yingbo Li

Abstract—**Multiview clustering explores complementary information among distinct views to enhance clustering performance under the assumption that all samples have complete information in all available views. However, this assumption does not hold in many real applications, where the information of some samples in one or more views may be missing, leading to partial multiview clustering problems. In this case, significant performance degeneration is usually observed. A collection of partial multiview clustering algorithms has been proposed to address this issue and most treat all different views equally during clustering. In fact, because different views provide features collected from different angles/feature spaces, they might play different roles in the clustering process. With the diversity of different views considered, in this study, a novel adaptive method is proposed for partial multiview clustering by automatically adjusting the contributions of different views. The samples are divided into complete and incomplete sets, while a joint learning mechanism is established to facilitate the connection between them and thereby improve clustering performance. More specifically, the method is characterized by a joint optimization model comprising two terms. The first term mines the underlying cluster structure from both complete and incomplete samples by adaptively updating their importance in all available views. The second term is designed to group all data with the aid of the cluster structure modeled in the first term. These two terms seamlessly integrate the complementary information among multiple views and enhance the performance of partial multiview clustering. Experimental results on real-world datasets illustrate the effectiveness and efficiency of our proposed method.**

Index Terms—**Partial multiview clustering, graph combination, adaptive weights.**

I. INTRODUCTION

With the rapid advancement in data-collection techniques, multiview data have become popular in various applications [36], [55]. As a new unsupervised method, multiview

clustering has been demonstrated as promising for research on the grouping structure of multiview data [2], [4], [46], [26]. Despite the numerous studies on multiview data, critical issues remain in real-world scenarios when some of the data from some views are inaccessible. This causes most of the existing methods to degenerate or even fail. This type of dataset usually contains two types of samples: those with complete information from all available views, and those with information missing in some of the views, as shown in Fig. 1. In general, incomplete data make multiview clustering a challenging problem. Even in some real applications, only the incomplete data are available in the dataset, which further complicates the clustering operation. In the literature, this type of scenario is often referred to as multiview clustering on partial[18], weakly-paired [17], incomplete [54], or semi-paired [35] data.

Fig. 1: Partial image-text dataset, which contains complete and incomplete data.

Many examples exist of partial multiview clustering, where data from different views are collected, processed, and stored independently. For example, in social platforms such as Facebook and Flickr, numerous digital images are uploaded to the website, but only some of them are annotated by users.

To deal with the partial multiview clustering problem, a series of approaches have been proposed in previous studies. One of the most intuitive strategies is simply removing the samples that have missing view information. This enables any existing multiview clustering method to be applied to the remaining samples with complete information. These methods completely ignore data with partial views and fails use all information available in the dataset. To address this issue, various completion methods have been proposed to fill the missing values based on certain predefined criteria. Nevertheless, no guarantee exists in terms of the fidelity of the information that has been added. In addition, multiple kernel- or graph-based learning techniques have been utilized to tackle multiview clustering from a different perspective. However, these methods are often computational demanding. Their poor efficiency limits their practical value in real-
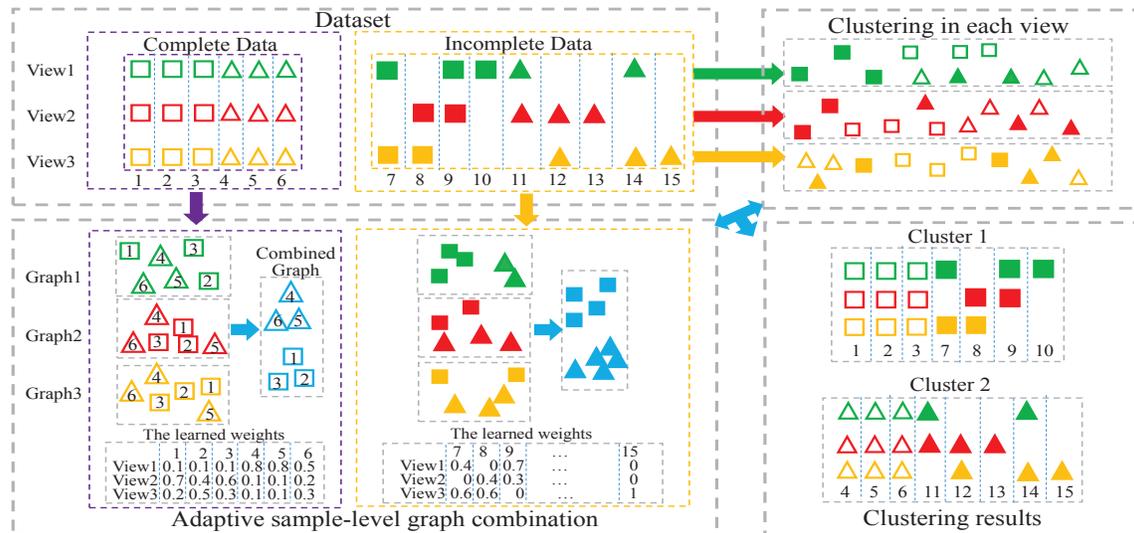
Fig. 2: Framework of the proposed method. A three-view dataset is considered as an example. To solve the partial multiview clustering problem, we project all partial data, including complete and incomplete data, into a unified clustering framework.

world applications. More recently, latent subspace methods have shown great promise in addressing the partial multiview clustering problem. Unfortunately, most of these methods treat all views equally during clustering, thus failing to consider the distinct expressive ability of different views. More importantly, the complete and incomplete data are often treated equally by most of these existing methods, which is undesirable.

In fact, different views and samples likely offer unique contributions to the partial multiview clustering task. Based on the example previously mentioned, annotation tags of pictures provided by users of social media are often quite brief and fail to fully represent the content of images. This means that more comprehensive information remains embedded in images, and thus image features and annotation tags will contribute differently to the clustering task. In addition, these differences might also be sample specific, as some images are assigned more detailed and meaningful tags. To tackle this problem, Shao et al. [33] proposed assigning different weights to partial samples and complete samples. Prior to performing clustering, they fixed a scaler assigned to all complete samples while setting another weighting factor for incomplete ones. However, the degree of freedom allowed in the clustering process was very limited, as the authors failed to consider sample-specific contributions from different views. When the image-text data in Fig. 1 is considered as an example, beyond the different levels of importance between complete and incomplete data, the first sample in a complete dataset cannot provide clear semantics based on its visual content. However, the corresponding tags are useful for identifying its cluster (i.e., the annotation tags are more informative than the visual content of the sample). By contrast, the visual content of a second sample is more informative than the annotation tags. Thus, a more reasonable approach is to establish a flexible framework in which an independent weighting factor for each view of each sample can be assigned and adjusted adaptively in the clustering process.

Accordingly, we propose an adaptive sample-level graph combination for partial multiview clustering (ASGC-PMVC) to cluster partial multiview samples based on sample-specific contributions from different views learned automatically (see Fig. 2). Consider a three-view dataset as an example with three views marked in green, red, and yellow, respectively. Without loss of generality, the dataset contains both complete and incomplete data with one or two missing views, as shown in the upper-left box of Fig. 2. Different shapes indicate data from different clusters, whereas the hollow and solid shapes are incorporated to differentiate the complete and incomplete samples. ASGC-PMVC is characterized by a joint optimization model consisting of two terms. The first term exploits the underlying cluster structure from the combined graphs of the complete or incomplete samples by evaluating the importance of each sample from each view. As shown in Fig. 2, the lower-left box illustrates the graph combination process of the complete or incomplete data. In the left purple box, the combined graph of complete samples can be constructed by automatically determining the importance of each view, where the weights of importance are marked under the graph combination. For example, the weights of the first sample in the three views are 0.1, 0.7, and 0.2, indicating that the second view is more important than the other views. Similarly, the combined graph of incomplete data is depicted in the right yellow box. The second term groups the entire partial multiview data within each view, which is illustrated in the upper-right box of Fig. 2. Samples with information loss in a view are not used in the clustering process of this view. The relation between these two terms is interpenetrative, interactive, and mutual-compensatory. We expect to obtain the clustering result by jointly minimizing these two terms, as shown in the lower right box. To summarize, the proposed method has the following advantages:

- ASGC-PMVC automatically determines the importance of each considered sample in each view and emphasizes

the importance of complete data.

- ASGC-PMVC learns the common subspace with the adaptive graph fusion, which can seamlessly integrate the complementary and consistent information from multiple views and enhance the clustering performance on partial multiview data.
- ASGC-PMVC can be efficiently solved, where experimental results on real-world datasets demonstrate the superiority of ASGC-PMVC over the baselines.

The remainder of this paper is organized as follows. In Section II, we briefly introduce and discuss several related works. Section III and IV present the proposed model and corresponding optimization algorithm, respectively. Experimental results are reported and discussed in Section V. Concluding remarks are provided in Section VI.

II. RELATED WORK

In this section, we briefly review existing studies on partial multiview clustering and adaptive weights learning for the multiview clustering task.

A. Partial Multiview Clustering

Multiview clustering deals with data represented by features from multiple domains and aims to enhance clustering performance [2], [36], [53]. An ideal situation is one in which each data sample has complete information in all views. However, in many real applications, information loss in some views may be inevitable. The existence of these partial views makes multiview clustering a challenging task, and the potential pitfalls of existing multiview clustering methods are expected.

Many different strategies have been proposed to handle the partial multiview clustering problem. A very intuitive method is to use complete data only, but this fails to utilize all the data. In addition, this type of simple strategy cannot be applied to the clustering of partial multiview samples, which are often encountered in the testing stage. An alternative approach is to complete the missing information based on certain assumptions about the data distribution with the aid of filling techniques. Some existing methods [33], [7], [21] simply fill the missing information by averaging over all the samples. These methods then learn the common representations for the multiple views. However, estimation missing information itself is a challenging task and these methods fail to obtain satisfactory accuracy, particularly with heterogeneous real-world datasets. More sophisticated algorithms have been proposed to improve the quality of missing information completion. Among them, Wang et al. [40] proposed exploring multi-view data generation for the clustering task based on generative adversarial networks [58], [30]. However, the reliability of the completed information remains questionable and this approach may also introduce noisy information into the data during completion process, which may bias the clustering process. The performance of these methods can further degrade when datasets are handled that have a considerable amount of missing information [43]. Advanced kernel or graph-based algorithms have also been developed to tackle the partial multiview clustering problem [1], [10], [22], [32], [34], [39], [49], [59],

[44], [51], [3], [23], [42]. These methods usually set absence rows and columns in kernels and graphs where information is missing and adopt multiple kernel learning techniques on all samples. Despite their promising performances, these methods suffer from high computational complexity, especially when dealing with large datasets [25].

Recently, some subspace learning methods have been proposed for partial multiview clustering. For example, canonical correlation analysis-based algorithms have been designed to incorporate paired and unpaired data [14], [15], [24], [52], [57] for a two-view problem. Eaton et al. [6] presented a constrained propagation method with incomplete mapping. However, their assumptions under multiple views, based on which the constraints are established, do not always hold true in real-world applications. Li and Chen [19] proposed a shared Gaussian process latent variable model, which defines the distribution in advance for a probabilistic graphical model. Yang et al. [48] proposed learning a common subspace representation based on sparse low-rank constraints for multiview dimension reduction. In addition, joint matrix factorization methods [20], [21], [31] have been adopted for partial multiview clustering [18] and are further equipped with must-link and cannot-link constraints [60] or other local manifold regularizations [45], [29], [28], [54], [50], [11], [38], [47], [56], [37], [43]. A common limitation of most of these methods is that they treat all views and samples equally, whereas different views of different samples may play distinct roles in the clustering task. As previously mentioned, a more reasonable method is to introduce a flexible sample-specific weighting scheme to highlight the important information. In the following section, we briefly review the existing weighting schemes that have been successfully applied to multiview clustering problems as well as their limitations. These have motivated us to develop the proposed method.

B. Adaptive Weights Learning

Successful pioneer studies have been conducted to adopt a weighting scheme in the multiview clustering process [27], [12], [13]. However, an obvious limitation of these methods is that they can deal only with complete multiview data and fail to consider variations in importance at the sample level. Furthermore, Shao et al. [33] proposed to treat complete and incomplete samples differently for clustering. Unfortunately, the weights for complete and incomplete samples are fixed prior to clustering, and their method fails to facilitate sample-specific weighting and realize adaptive weight adjustment for optimal clustering performance.

To resolve these issues, we propose a partial multiview clustering method. The method is designed to exploit a common latent space for multiview clustering. The importance of different views for different samples in the proposed method can be automatically updated during the learning process.

III. FRAMEWORK OF ADAPTIVE SAMPLE-LEVEL GRAPH COMBINATION FOR PARTIAL MULTIVIEW CLUSTERING

Complete multiview data are often expensive and time-consuming to obtain. Thus, we plan to cluster partial data

containing considerable incomplete data. We first must list some primary notations.
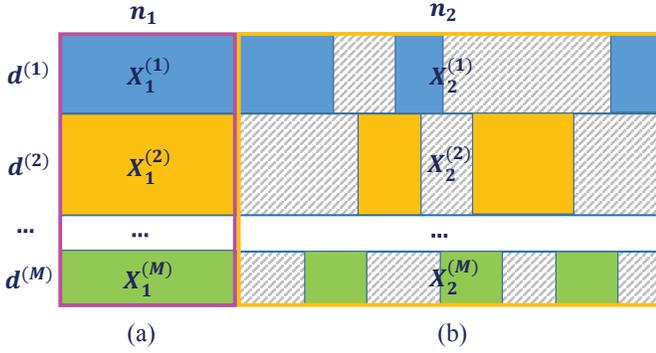
A. Primary Notations

Fig. 3: The partial dataset is composed of (a) complete and (b) incomplete parts. Note that $n_1 = 0$ means that all samples suffer from missing views, whereas $n_2 = 0$ means that all samples have complete views.

In the partial setup of M views, some view information of N samples is missing. Thus, all samples can be divided into two parts (i.e., complete and incomplete parts), as indicated by purple and yellow boxes in Fig. 3, respectively. Let $\mathcal{X} = \{\mathcal{X}_1, \mathcal{X}_2\}$ be the dataset. $\mathcal{X}_1$ denotes the set of n_1 complete samples appearing in all views, where each sample is represented as a set of feature vectors in each view (i.e., the data matrix $X_1^{(m)} \in \mathbb{R}^{d^{(m)} \times n_1}$, where $d^{(m)}$ is the number of features in the m-th view). $\mathcal{X}_2$ indicates the set of n_2 samples with missing views (i.e., the data matrix $X_2^{(m)} \in \mathbb{R}^{d^{(m)} \times n_2}$). If the j-th sample in the m-th view is available, then it is represented by the corresponding $d^{(m)}$ features. Otherwise, the m-th view of this sample is missing, which indicates that the sample is not useful in this view, and the missing values are set zeroes. Let $X^{(m)} = [X_1^{(m)} \ X_2^{(m)}] \in \mathbb{R}^{d^{(m)} \times N}$, where $N = n_1 + n_2$ denotes the total number of samples. We aim to group N samples into k clusters.

To record the presence of variables in the m-th view, we introduce a diagonal matrix $W^{(m)} \in \mathbb{R}^{N \times N}$ for each view. If $(W^{(m)})_{jj} = 1$, then the j-th sample is available in the m-th view. If the j-th sample is missing (i.e., $(W^{(m)})_{jj} = 0$), then this sample will be ignored. As the dataset has been divided into complete and incomplete parts ($\mathcal{X}_1$ and $\mathcal{X}_2$), the diagonal matrices of the corresponding samples are $W_1^{(m)} \in \mathbb{R}^{n_1 \times n_1}$ and $W_2^{(m)} \in \mathbb{R}^{n_2 \times n_2}$. Because the samples in $\mathcal{X}_1$ are complete in all views, $W_1^{(m)} = I_{n_1}$ is an identity matrix for each view. The primary notations are summarized in Table I.

B. Adaptive Sample-level Graph Combination for Complete Data $\mathcal{X}_1$ and Incomplete Data $\mathcal{X}_2$

As the dataset has been divided into complete and incomplete parts, adaptive sample-level graphs are constructed based on these two parts. To represent the complete data $\mathcal{X}_1$ sufficiently from all views (note that these views may be heterogeneous), we introduce a mapping function from the

original heterogeneous feature spaces into a similarity space. Moreover, this mapping function is expected to characterize the importance of different samples in different views. More specifically, for each view, a nearest-neighbor graph $G_1^{(m)} \in \mathbb{R}^{n_1 \times n_1}$ for complete data $X_1^{(m)}$ can be constructed by setting each complete sample as a vertex and the similarity between two samples as an edge [8]. After obtaining M nearest-neighbor graphs $\{G_1^{(m)}\}_{m=1}^M$, one for each view, we attempt to combine them to construct a graph by considering the importance of each sample in each view.

Furthermore, a set of variables $\{\boldsymbol{\lambda}_1^{(m)}\}_{m=1}^M$ is introduced to characterize the adaptive sample-level weights. Each $\boldsymbol{\lambda}_1^{(m)}$ is an n_1-dimensional vector, where $(\boldsymbol{\lambda}_1^{(m)})_j$ indicates the importance of the j-th complete sample in the m-th view. In other words, each sample has M weights, and each weight indicates the importance of the current sample in the corresponding view. Without loss of generality, these M weights for each sample are further required to be non-negative and normalized. Accordingly, we combine the M graphs by $\sum_{m=1}^M diag(\boldsymbol{\lambda}_1^{(m)})G_1^{(m)}W_1^{(m)}$, where $diag(\boldsymbol{\lambda}_1^{(m)})$ is the diagonal matrix of $\boldsymbol{\lambda}_1^{(m)}$ and $W_1^{(m)}$ is the diagonal matrix indicating the available samples. To guarantee symmetry, we set:

$$\mathbb{G}_1 = \frac{1}{2} \sum_{m=1}^M \left(diag(\boldsymbol{\lambda}_1^{(m)})G_1^{(m)}W_1^{(m)} + W_1^{(m)}G_1^{(m)}diag(\boldsymbol{\lambda}_1^{(m)}) \right)$$

$$(0 \leq \boldsymbol{\lambda}_1^{(m)} \leq 1, \forall m, \sum_{m=1}^M (\boldsymbol{\lambda}_1^{(m)})_j = 1, \forall j).$$

Note that for complete data, $W_1^{(m)}$ is the identity matrix and can be omitted. This graph combination strategy is called ASGC.

Using the combined graph $\mathbb{G}_1$, we can mine the cluster structure using different techniques such as spectral clustering, non-negative matrix factorization (NMF), or Sym-NMF. As Sym-NMF often outperforms other methods [16], identification of the clustering structure from complete data can be formulated as:

$$\min_{\{\boldsymbol{\lambda}_1^{(m)}\}_{m=1}^M, H_1} \left\| \mathbb{G}_1 - H_1^T H_1 \right\|_F^2 \qquad (1)$$

$$\text{s.t.} \ \ H_1 \geq 0, 0 \leq \boldsymbol{\lambda}_1^{(m)} \leq 1, \forall m, \sum_{m=1}^M (\boldsymbol{\lambda}_1^{(m)})_j = 1, \forall j.$$

$H_1 \in \mathbb{R}^{k \times n_1}$ is the desired cluster-sample matrix with k clusters.

Similarly, the graph $\mathbb{G}_2$ is combined to represent incomplete data $\mathcal{X}_2$ sufficiently from all views. Note that there are missing views for the incomplete data; if $(W_2^{(m)})_{jj} = 0$, which indicates that the j-th sample is ignored in the m-th view, the weight $(\boldsymbol{\lambda}_2^{(r)})_j$ will be set to zero. The graph $\mathbb{G}_2$ is constructed by:

$$\mathbb{G}_2 = \frac{1}{2} \sum_{m=1}^M \left(diag(\boldsymbol{\lambda}_2^{(m)})G_2^{(m)}W_2^{(m)} + W_2^{(m)}G_2^{(m)}diag(\boldsymbol{\lambda}_2^{(m)}) \right)$$

$$(0 \leq \boldsymbol{\lambda}_2^{(m)} \leq 1, \forall m, \sum_{m=1}^M (\boldsymbol{\lambda}_2^{(m)})_j = 1, \forall j).$$

TABLE I: Summary of notations

Notation	Size	Description
n_1		Number of complete samples
n_2		Number of incomplete samples
N		Total number of samples $N = n_1 + n_2$
M		Number of views
k		Number of clusters
$d^{(m)}$		Dimension of features in the m-th view
$X_1^{(m)}$	$d^{(m)} \times n_1$	Data matrix of complete samples in the m-th view
$X_2^{(m)}$	$d^{(m)} \times n_2$	Data matrix of incomplete samples in the m-th view
$X^{(m)}$	$d^{(m)} \times N$	Data matrix of the m-th view consisting of $X_1^{(m)}$ and $X_2^{(m)}$, $X^{(m)} = [X_1^{(m)} \ X_2^{(m)}]$
$W^{(m)}$	$N \times N$	A diagonal matrix, where $(W^{(m)})_{jj}$ indicates the availability of the j-th sample in the m-th view
$G_1^{(m)}$	$n_1 \times n_1$	Nearest neighbor graph constructed of complete data $X_1^{(m)}$ in the m-th view
$\mathbb{G}_1$	$n_1 \times n_1$	Graph combined by $G_1^{(m)}$ in all M views
$G_2^{(m)}$	$n_2 \times n_2$	Nearest neighbor graph constructed of incomplete data $X_2^{(m)}$ in the m-th view
$\mathbb{G}_2$	$n_2 \times n_2$	Graph combined by $G_2^{(m)}$ in all M views
$Z^{(m)}$	$d^{(m)} \times k$	Basis matrix of the m-th view
H_1	$k \times n_1$	Latent representation of the complete data X_1
H_2	$k \times n_2$	Latent representation of the incomplete data X_2
H	$k \times N$	$H = [H_1 \ H_2]$

Sym-NMF on the incomplete data can be formulated as:

$$\min_{\{\boldsymbol{\lambda}_2^{(m)}\}_{m=1}^M, H_2} \left\| \mathbb{G}_2 - H_2^T H_2 \right\|_F^2 \qquad (2)$$

$$\text{s.t.} \quad H_2 \geq 0, 0 \leq \boldsymbol{\lambda}_2^{(m)} \leq 1, \forall m, \sum_{m=1}^M (\boldsymbol{\lambda}_2^{(m)})_j = 1, \forall j.$$

$H_2 \in \mathbb{R}^{k \times n_2}$ is the cluster indicator of $\mathcal{X}_2$.

Thus, (1) and (2) can be combined as $\sum_{i=1}^2 \alpha_i \left\| \mathbb{G}_i - H_i^T H_i \right\|_F^2$ to determine the clustering structure H_i from complete data $\mathcal{X}_1$ and incomplete data $\mathcal{X}_2$, where the parameter α_i is used to balance the contribution of complete and incomplete samples.

C. Proposed ASGC-PMVC Method

Each view contains samples with complete or incomplete information (i.e., $X^{(m)} = [X_1^{(m)} \ X_2^{(m)}]$). To perform partial multiview clustering on the available data $\{X^{(m)} W^{(m)}\}_{m=1}^M$ with the aid of ASGC, the proposed ASGC-PMVC is formulated as follows:

$$\min_{\Theta} \sum_{i=1}^2 \alpha_i \left\| \mathbb{G}_i - H_i^T H_i \right\|_F^2$$

$$+ \sum_{m=1}^M \eta^{(m)} ||(X^{(m)} - Z^{(m)} H) W^{(m)}||_F^2 \qquad (3)$$

$$\text{s.t.} \quad H_i, Z^{(m)} \geq 0, 0 \leq \boldsymbol{\lambda}_i^{(m)} \leq 1, \forall i, m,$$

$$\sum_{m=1}^M (\boldsymbol{\lambda}_i^{(m)})_j = 1, \forall i, j$$

where $\Theta = \{\{Z^{(m)}\}_{m=1}^M, \{\{\boldsymbol{\lambda}_i^{(m)}\}_{m=1}^M, H_i\}_{i=1}^2\}$ is the set of desired variables and $W^{(m)} \in \mathbb{R}^{N \times N}$ is the diagonal matrix that ignores the missing samples in the m-th view. The first term determines the clustering structure H_i according to the adaptive strategy. The goal of the second term is to identify the cluster structure $H = [H_1 \ H_2] \in \mathbb{R}^{k \times N}$ of each view $X^{(m)}$ with the available samples. To cluster the data of each view, we introduce the NMF model to evaluate the loss between original data $X^{(m)}$ and the reconstructed data $Z^{(m)} H$ of the available

samples in each view. $Z^{(m)} \in \mathbb{R}^{d^{(m)} \times k}$ is the basis factor of the m-th view. The cluster structure H_1 of the complete data can be used to guide the clustering process on the incomplete data. Moreover, H_2 obtained from the incomplete data can be used to generate a better estimation of H_1. In this case, the complete and incomplete data can be sufficiently integrated to achieve a better clustering result. The different views share a common $H = [H_1 \ H_2]$ to ensure consistency. The basis factor $Z^{(m)}$ and the corresponding cluster indicator H enforce all samples to be smoothly gathered in the clustering process. The parameter $\eta^{(m)}$ can balance the contribution of different views.

D. Analysis

The proposed framework can handle an extreme case with no complete samples (i.e., $n_1 = 0$). In this case, the first term in the proposed model (3) is reduced to $\left\| \mathbb{G}_2 - H_2^T H_2 \right\|_F^2$. If $n_2 = 0$, which indicates that all the samples have complete views, (3) is reduced to $\left\| \mathbb{G}_1 - H_1^T H_1 \right\|_F^2$. In addition, the second term in (3) can be flexibly and conveniently replaced with other loss functions, such as the hinge, and logistic loss functions.

The proposed method is different from the existing methods in the following ways. 1) It can automatically learn the adaptive sample-level weight $\boldsymbol{\lambda}_i^{(m)}$ for each view to determine the importance of each considered sample. 2) All the samples are divided into complete and incomplete parts, and the parameters α_1 and α_2 are used to balance the contribution of complete and incomplete data, respectively. 3) $\eta^{(m)}$ is introduced to represent the importance of each view, and its value can be set according to $\boldsymbol{\lambda}_i^{(m)}$, which is illustrated in the experiments described in Sections V.D and V.E. 4) It learns the common subspace with the adaptive graph fusion and can then seamlessly integrate the complementary and consistent information from multiple views for partial multiview data.

IV. OPTIMIZATION ALGORITHM

To address the joint nonconvex optimization problem of ASGC-PMVC, we utilize a numerical scheme to alternately update each of the variables by fixing the others.

More specifically, (3) can be rewritten as:

$$\min_{\Theta} \sum_{i=1}^{2} \alpha_i \left\| \frac{1}{2} \sum_{m=1}^{M} \left(W_i^{(m)} G_i^{(m)} \Lambda_i^{(m)} + \Lambda_i^{(m)} G_i^{(m)} W_i^{(m)} \right) \right.$$
$$\left. - H_i^T H_i \right\|_F^2 + \sum_{m=1}^{M} \eta^{(m)} \| (X^{(m)} - Z^{(m)} H) W^{(m)} \|_F^2 \quad (4)$$

$$\text{s.t. } Z^{(m)}, H_i \geq 0, 0 \leq \Lambda_i^{(m)} \leq 1, \forall m, \sum_{m=1}^{M} \Lambda_i^{(m)} = I_i, \forall i$$

where $\Lambda_i^{(m)} = diag(\boldsymbol{\lambda}_i^{(m)})$ and $I_i \in \mathbb{R}^{n_i \times n_i}$ is an identity matrix. To solve the constraints in (4), we consider its augmented Lagrangian function $\mathcal{O}$, which can be derived as:

$$\mathcal{O} \equiv \sum_{i=1}^{2} \alpha_i \left(\left\| \frac{1}{2} \widehat{G}_i \widehat{\Lambda}_i + \frac{1}{2} Y_i - J_i^T H_i \right\|_F^2 \right.$$
$$+ \beta_i \| J_i - H_i \|_F^2 + 2 < \Gamma_i, J_i - H_i >$$
$$+ \gamma_i \| E_i \widehat{\Lambda}_i - I_i \|_F^2 + 2 < \Psi_i, E_i \widehat{\Lambda}_i - I_i > \quad (5)$$
$$+ \theta_i \| \widehat{\Lambda}_i^T \widehat{G}_i^T - Y_i \|_F^2 + 2 < \Phi_i, \widehat{\Lambda}_i^T \widehat{G}_i^T - Y_i >$$
$$\left. + \{ \delta_+(\widehat{\Lambda}_i^{(m)}) \}_{m=1}^{M} + \delta_+(H_i) \right)$$
$$+ \sum_{m=1}^{M} \left(\eta^{(m)} \| (X^{(m)} - Z^{(m)} H) W^{(m)} \|_F^2 + \delta_+(Z^{(m)}) \right)$$

where Y_i and J_i are auxiliary variables to fit $\widehat{\Lambda}_i^T \widehat{G}_i^T$ and H_i, respectively. $\beta_i, \gamma_i, \theta_i > 0$ are the penalty parameters, and Γ_i, Ψ_i, Φ_i are the Lagrange multipliers. To optimize all $\{\Lambda_i^{(m)}\}_{m=1}^{M}$ together, we set $\widehat{G} = \left[(W_i^{(1)} G_i^{(1)}) \ (W_i^{(2)} G_i^{(2)}) \ \cdots (W_i^{(M)} G_i^{(M)}) \right]$, $\widehat{\Lambda} = \begin{bmatrix} \Lambda_i^{(1)} \\ \Lambda_i^{(2)} \\ \cdots \\ \Lambda_i^{(M)} \end{bmatrix}$, $E_i = [I_i \ I_i \cdots I_i]$. δ_+ represents the delta function that provides $+\infty$ to the negative values. This guarantees the non-negativity of the optimal solution because the presence of negative element leads to an infinitely large objective function value.

A. Minimizing $\mathcal{O}$ over $\widehat{\Lambda}_i$

Given $(H_i)^{\tau}, (J_i)^{\tau}, (Y_i)^{\tau}, (\Psi_i)^{\tau}, (\Phi_i)^{\tau}$ (τ is the iteration number), $\widehat{\Lambda}_i$ can be solved by optimizing the following problem:

$$\min_{\widehat{\Lambda}_i} \mathcal{O}\left(\widehat{\Lambda}_i\right) \equiv \left\| \frac{1}{2} \widehat{G}_i \widehat{\Lambda}_i + \frac{1}{2} (Y_i)^{\tau} - (J_i^{\tau})^T (H_i)^{\tau} \right\|_F^2$$
$$+ \gamma_i \| E_i \widehat{\Lambda}_i - I_i \|_F^2 + 2 < (\Psi_i)^{\tau}, E_i \widehat{\Lambda}_i - I_i >$$
$$+ \theta_i \left\| \widehat{\Lambda}_i^T \widehat{G}_i^T - (Y_i)^{\tau} \right\|_F^2 + \delta_+\left(\widehat{\Lambda}_i\right)$$
$$+ 2 < (\Phi_i)^{\tau}, \widehat{\Lambda}_i^T \widehat{G}_i^T - (Y_i)^{\tau} > \quad (6)$$

The updating rule is:

$$(\widehat{\Lambda}_i)^{\tau+1/2} := \left((\frac{1}{2} + 2\theta_i) \widehat{G}_i^T \widehat{G}_i + 2\gamma_i E_i^T E_i \right)^{-1}$$
$$\left(\widehat{G}_i^T \left(-\frac{1}{2} (Y_i)^{\tau} + 2\theta_i ((Y_i)^{\tau})^T \right. \right.$$
$$+ ((J_i)^{\tau})^T (H_i)^{\tau} - 2 ((\Phi_i)^{\tau})^T)$$
$$\left. + 2\gamma_i E_i^T - 2 E_i^T (\Psi_i)^{\tau} \right)$$
$$(\widehat{\Lambda}_i)^{\tau+1} := Proj((\widehat{\Lambda}_i)^{\tau+1})_+ \quad (7)$$

where $Proj((\widehat{\Lambda}_i)^{\tau+1})_+$ indicates that if $(\widehat{\Lambda}_i)^{\tau+1} < 0$, then set $(\widehat{\Lambda}_i)^{\tau+1} = 0$.

B. Minimizing $\mathcal{O}$ over Y_i

Given $(\widehat{\Lambda}_i)^{\tau+1}, (H_i)^{\tau}, (J_i)^{\tau}, (\Phi_i)^{\tau}$, Y_i can be solved by optimizing the following problem:

$$\min_{Y_i} \mathcal{O}(Y_i) \equiv \left\| \frac{1}{2} \widehat{G}_i (\widehat{\Lambda}_i)^{\tau+1} + \frac{1}{2} Y_i - ((J_i)^{\tau})^T (H_i)^{\tau} \right\|_F^2$$
$$+ \theta_i \| ((\widehat{\Lambda}_i)^{\tau+1})^T \widehat{G}_i^T - Y_i \|_F^2$$
$$+ 2 < (\Phi_i)^{\tau}, ((\widehat{\Lambda}_i)^{\tau+1})^T \widehat{G}_i^T - Y_i > \quad (8)$$

By setting the gradient of $\mathcal{O}(Y_i)$ to zero, we obtain the new Y_i as follows:

$$(Y_i)^{\tau+1} := \left(1/(\frac{1}{2} + 2\theta_i) \right) \left(-\frac{1}{2} \widehat{G}_i (\widehat{\Lambda}_i)^{\tau+1} + 2(\Phi_i)^{\tau} \right.$$
$$\left. + ((J_i)^{\tau})^T (H_i)^{\tau} + 2\theta_i ((\widehat{\Lambda}_i)^{\tau+1})^T \widehat{G}_i^T \right) \quad (9)$$

C. Minimizing $\mathcal{O}$ over J_i

Given $(\widehat{\Lambda}_i)^{\tau+1}, (Y_i)^{\tau+1}, (H_i)^{\tau}, (\Gamma_i)^{\tau}$, J_i can be solved by optimizing the following problem:

$$\min_{J_i} \mathcal{O}(J_i) \equiv \beta_i \| J_i - (H_i)^{\tau} \|_F^2 + 2 < (\Gamma_i)^{\tau}, J_i - (H_i)^{\tau} >$$
$$+ \left\| ((H_i)^{\tau})^T J_i - \frac{1}{2} ((Y_i)^{\tau+1})^T - \frac{1}{2} ((\widehat{\Lambda}_i)^{\tau+1})^T \widehat{G}_i^T \right\|_F^2 \quad (10)$$

We directly obtain its gradient, and the updating rule is:

$$(J_i)^{\tau+1} := \left(2(H_i)^{\tau} ((H_i)^{\tau})^T + 2\beta_i I_k \right)^{-1}$$
$$\left((H_i)^{\tau} ((Y_i)^{\tau+1})^T + (H_i)^{\tau} ((\widehat{\Lambda}_i)^{\tau+1})^T \widehat{G}_i^T \right.$$
$$\left. + 2\beta_i (H_i)^{\tau} - 2(\Gamma_i)^{\tau} \right) \quad (11)$$

where $I_k \in \mathbb{R}^{k \times k}$.

D. Minimizing $\mathcal{O}$ over H_i

Given $(\widehat{\Lambda}_i)^{\tau+1}, (J_i)^{\tau+1}, (Y_i)^{\tau+1}, (\Gamma_i)^{\tau}, (Z^{(m)})^{\tau}$ for each view, H_i can be solved by optimizing the following problem:

$$\min_{H_i} \mathcal{O}(H_i) \equiv \alpha_i \left(\| A_i - ((J_i)^{\tau+1})^T H_i \|_F^2 + \beta_i \| (J_i)^{\tau+1} - H_i \|_F^2 \right.$$
$$+ 2 < (\Gamma_i)^{\tau}, (J_i)^{\tau+1} - H_i > + \delta_+(H_i))$$
$$+ \sum_{m=1}^{M} \eta^{(m)} \| (X_i^{(m)} - (Z^{(m)})^{\tau} H_i) W^{(m)} \|_F^2 \quad (12)$$

where $A_i = \frac{1}{2} \widehat{G}_i (\widehat{\Lambda}_i)^{\tau+1} + \frac{1}{2} (Y_i)^{\tau+1}$. H_i can be updated as:

$$(H_i)^{\tau+1/2} := \Bigg(\alpha_i \big((J_i)^{\tau+1}((J_i)^{\tau+1})^T + \beta_i I_k\big)$$
$$+ \sum_{m=1}^{M} \eta^{(m)}((Z^{(m)})^\tau)^T (Z^{(m)})^\tau \Bigg)^{-1}$$
$$\Bigg(\sum_{m=1}^{M} \eta^{(m)}((Z^{(m)})^\tau)^T X_i^{(m)} W^{(m)}$$
$$+ \alpha_i \big((J_i)^{\tau+1} A_i + \beta_i (J_i)^{\tau+1} + (\Gamma_i)^\tau\big) \Bigg)$$
$$(H_i)^{\tau+1} := Proj((H_i)^{\tau+1})_+$$

E. Updating Γ_i, Ψ_i and Φ_i

$$(\Gamma_i)^{\tau+1} := (\Gamma_i)^\tau + \beta_i((J_i)^{\tau+1} - (H_i)^{\tau+1})$$
$$(\Psi_i)^{\tau+1} := (\Psi_i)^\tau + \gamma_i \left(E_i(\widehat{\Lambda})^{\tau+1} - I_i \right) \quad (14)$$
$$(\Phi_i)^{\tau+1} := (\Phi_i)^\tau + \theta_i \left(((\widehat{\Lambda}_i)^{\tau+1})^T \widehat{G}_i^T - (Y_i)^{\tau+1} \right)$$

F. Minimizing $\mathcal{O}$ over $Z^{(m)}$

Given $(H)^{\tau+1} = [(H_1)^{\tau+1}(H_2)^{\tau+1}]$, the computations of $\{Z^{(m)}\}_{m=1}^M$ for each view do not depend on each other. Therefore, (5) is reduced to:

$$\min_{Z^{(m)}} \mathcal{O}(Z^{(m)}) \equiv \eta^{(m)} ||(X^{(m)} - Z^{(m)}(H)^{\tau+1})W^{(m)}||_F^2$$
$$+ \delta_+(Z^{(m)}) \quad (15)$$

It can be solved as follows:

$$(Z^{(m)})^{\tau+1/2} := X^{(m)} W^{(m)} ((H)^{\tau+1})^T$$
$$\left((H)^{\tau+1} W^{(m)} ((H)^{\tau+1})^T \right)^{-1}$$
$$(Z^{(m)})^{\tau+1} := Proj\left((Z^{(m)})^{\tau+1/2} \right)_+ \quad (16)$$

Note that the computations of $Z^{(m)}$ for each view do not depend on each other, and the computations for all M views can be run in parallel.

The overall process of ASGC-PMVC is described in Algorithm 1.

G. Convergence Analysis

The proposed ASGC-PMVC is nonconvex. However, its convergence to a stationary point can still be established. For the model in (4), the proposed algorithm tackles the optimization problem in an alternative manner. More specifically, we solve one of $\{\widehat{\Lambda}_i, H_i\}_{i=1}^2$ and $\{Z^{(m)}\}_{m=1}^M$, while fixing the others. It can be demonstrated that the value of the objective function decreases if the minimization problem with respect to each individual variable can be solved properly.

To tackle the subproblem of $\widehat{\Lambda}_i$, which is a constrained convex optimization problem, we utilize the alternating direction

---

Algorithm 1: ASGC-PMVC

Input : Data matrix of the m-th view $\{X^{(m)}\}_{m=1}^M$, diagonal matrix $\{W^{(m)}\}_{m=1}^M$, where $W_{jj}^{(m)}$ indicates the availability of j-th sample in the m-th view, parameters $\{\alpha_i\}_{i=1}^2$ and $\{\eta^{(m)}\}_{m=1}^M$.

Output: Clustering results.

Construct nearest-neighbor graphs $\{G_1^{(m)}\}_{m=1}^M$ and $\{G_2^{(m)}\}_{m=1}^M$.

repeat
  for $i = 1 : 2$ **do**
    Update $\widehat{\Lambda}_i$ via (7).
    Update Y_i via (9).
    Update J_i via (11).
    Update H_i via (13).
    Update Γ_i, Ψ_i and Φ_i via (14).
  end
  Update $H = [H_1 \; H_2]$.
  for $m = 1 : M$ **do**
    Update $Z^{(m)}$ via (16).
  end
until *convergence*;
Obtain clustering results of the new representation H.
return

---

method of multipliers (ADMM). The following corresponding augmented Lagrangian function is considered at each step:

$$\left\| \frac{1}{2}\widehat{G}_i\widehat{\Lambda}_i + \frac{1}{2}Y_i - ((H_i)^\tau)^T (H_i)^\tau \right\|_F^2$$
$$+ \zeta_i \left\| \begin{bmatrix} E_i \\ \widehat{G}_i \end{bmatrix} \widehat{\Lambda}_i - \begin{bmatrix} I_i \\ Y_i^T \end{bmatrix} \right\|_F^2$$
$$+ 2\left\langle \begin{bmatrix} \Psi_i \\ \Phi_i^T \end{bmatrix}, \begin{bmatrix} E_i \\ \widehat{G}_i \end{bmatrix} \widehat{\Lambda} - \begin{bmatrix} I_i \\ Y_i^T \end{bmatrix} \right\rangle + \delta_+(\Lambda), (17)$$

where Y_i is an auxiliary variable and ζ_i is a parameter obtained by combining γ_i and θ_i. The global convergence of ADMM on a convex optimization problem is well established. Let F denote the objective function given in (17). We can then obtain the following inequality:

$$F\left((\widehat{\Lambda}_i)^{\tau+1}, (H_i)^\tau, \{(Z^{(m)})^\tau\}_{m=1}^M \right)$$
$$\leq F\left(\widehat{\Lambda}_i, (H_i)^\tau, \{(Z^{(m)})^\tau\}_{m=1}^M \right). \quad (18)$$

The subproblem with respect to H_i is nonconvex. However, we can still use ADMM to split the problem by introducing auxiliary variables J_i, and we can formulate the corresponding augmented Lagrangian as:

$$\alpha_i \left\| \frac{1}{2}\widehat{G}_i(\widehat{\Lambda}_i)^{\tau+1} + \frac{1}{2}(\widehat{\Lambda}_i)^{\tau+1}\widehat{G}_i - J_i^T H_i \right\|_F^2$$
$$+ \sum_{m=1}^{M} \eta^{(m)} \left\| \left(X_i^{(m)} - (Z^{(m)})^\tau (H_i)^\tau \right) W^{(m)} \right\|_F^2$$
$$+ \beta_i ||J_i - H_i||_F^2 + 2 < \Gamma_i, J_i - H_i > + \delta_+(H_i) \; (19)$$

Based on [9] [41], ADMM promises convergence to one of its stationary points. Thus, we have:

$$F\left((\widehat{\Lambda}_i)^{\tau+1}, (H_i)^{\tau+1}, \{(Z^{(m)})^{\tau}\}_{m=1}^M\right)$$
$$\leq F\left((\widehat{\Lambda}_i)^{\tau+1}, H_i, \{(Z^{(m)})^{\tau}\}_{m=1}^M\right). \qquad (20)$$

The subproblem of $Z^{(m)}$ is convex and can be easily solved by seeking solutions to linear systems, as given in (16). Thus, we have:

$$F\left((\widehat{\Lambda}_i)^{\tau+1}, (H_i)^{\tau+1}, \{(Z^{(m)})^{\tau+1}\}_{m=1}^M\right)$$
$$\leq F\left((\widehat{\Lambda}_i)^{\tau+1}, (H_i)^{\tau+1}, \{Z^{(m)}\}_{m=1}^M\right). \qquad (21)$$

Overall, by iteratively solving $\widehat{\Lambda}_i$, H_i and $\{Z^{(m)}\}_{m=1}^M$ using the scheme previously mentioned, the value of the objective function should monotonically decrease. Furthermore, as the objective function is bounded below, the proposed algorithm converges to a stationary point.

To reduce the overall computational complexity, we update $\widehat{\Lambda}_i$ and H_i according to the ADMM scheme only once at each iteration, which is equivalent to directly handling the combined augmented Lagrangian function, as follows:

$$\alpha_i \left\| \frac{1}{2}\widehat{G}_i\Lambda_i + \frac{1}{2}Y_i - J_i^T H_i \right\|_F^2$$
$$+ \sum_{m=1}^M \eta^{(m)} \left\| \left(X^{(m)} - (Z^{(m)})^{\tau}H\right)W^{(m)} \right\|_F^2$$
$$+ \beta_i \|J_i - H_i\|_F^2 + 2 < \Gamma_i, J_i - H_i > \qquad (22)$$
$$+ \zeta_i \left\| \begin{bmatrix} E_i \\ \widehat{G}_i \end{bmatrix} \widehat{\Lambda}_i - \begin{bmatrix} I_i \\ Y_i^T \end{bmatrix} \right\|_F^2$$
$$+ 2\left\langle \begin{bmatrix} \Psi_i \\ \Phi_i^T \end{bmatrix}, \begin{bmatrix} E_i \\ \widehat{G}_i \end{bmatrix} \widehat{\Lambda}_i - \begin{bmatrix} I_i \\ Y_i^T \end{bmatrix} \right\rangle + \delta_+(\widehat{\Lambda}_i) + \delta_+(H_i)$$

In addition, we can further relax the problem so that the penalty parameters of the two constraints with respect to $\widehat{\Lambda}_i$ can be different. We observed that the algorithm is convergent for all tested datasets.

H. Computational Complexity

In this section, we consider the computational cost of ASGC-PMVC. In (7), we calculate the inverse of $((\frac{1}{2} + 2\theta_i)\widehat{G}_i^T\widehat{G}_i + 2\gamma_i E_i^T E_i)$ with $\mathcal{O}(Mn_i)^3$ complexity. As the inverse is fixed throughout the ADMM iterations, computing the inverse in advance is sufficient . The complexities of the inverse in (11), (13), and (16) are all $\mathcal{O}(k^3)$. As the number of clusters k is usually very small, the inverse is not the dominant cost. k is always smaller than n_i, and thus the costs of updating $\widehat{\Lambda}_i$, Y_i, J_i, Ψ_i, and Φ_i are all $\mathcal{O}(Mn_i^3)$. The costs of updating $Z^{(m)}$, H_i, and Γ_i are $\mathcal{O}(kd^{(m)}\overline{n}^{(m)})$, $\mathcal{O}(\max(kd^{(m)}n_i, kn_i^2))$, and $\mathcal{O}(kn_i)$, respectively, where $\overline{n}^{(m)}$ is the number of samples without missing values in the m-th view. Overall, the computational cost of ASGC-PMVC for each iteration is $\mathcal{O}\left(\max\{Mn_i^3\}_{i=1}^2 + \max(\{kd^{(m)}\overline{n}^{(m)}\}_{m=1}^M, \{kn_i^2\}_{i=1}^2)\right)$. Note that n_i is the number of complete or incomplete samples represented with all views, and $\overline{n}^{(m)}$ is the number of samples represented in the m-th view; they are less than the number of all samples (N). Moreover, the update of $\widehat{\Lambda}_i$, J_i, Y_i, H_i for

the complete or incomplete parts can be run simultaneously, and $Z^{(m)}$ of different views can be updated in parallel because they are independent of each other in the optimization process. Thus, the complexities are not very high.

V. Experimental Results

A series of experiments was conducted to validate the performance of the proposed models.

A. Synthetic Experiment

To better understand the idea behind the combination of graphs on the complete data, we conducted an experiment on synthetic data. As shown in Fig. 4, 400 samples were represented in two views, and all the samples originated from four clusters, which are marked in the figure by different shapes. The number of complete and incomplete samples was 200 each, and they are illustrated by the hollow and solid shapes, respectively. Furthermore, 120 and the other 80 incomplete samples are represented in the first and second views, respectively. All the data were distributed in 2D space, and were generated by Gaussian distributions with different means and covariance matrices.
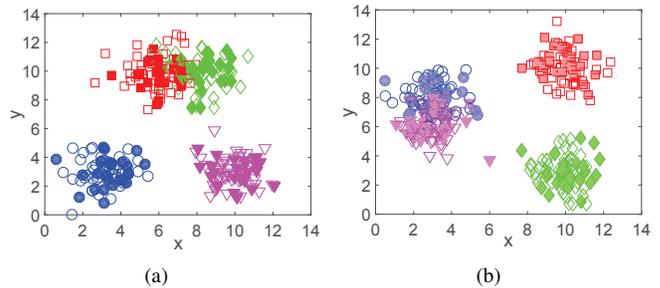
Fig. 4: Synthetic data in (a) View$_1$ and (b) View$_2$. The samples with the same shape belong to the same cluster.

In Fig. 4(a), some samples in the first view originating from the circle and triangle clusters have good separation, whereas the square and diamond data are not well separated. However, the distribution of data in the second view in Fig. 4(b) is in contrast to that in Fig. 4(a). As only two views existed in this dataset, if the j-th incomplete sample appeared in the first view, then its weight $(\lambda_2^{(1)})_j = 1$ and $(\lambda_2^{(2)})_j = 0$. Thus, the weight of each incomplete sample did not need to be learned, and we only had to learn the weights of the complete samples.

Fig. 6: Weights of the first view ($\lambda_1^{(1)}$) of each complete data for synthetic data.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2019.2952696, IEEE Transactions on Image Processing

9

Fig. 5: Graph constructed using the complete data (a) $G_1^{(1)}$, (b) $G_1^{(2)}$, (c) $\mathbb{G}_1$.

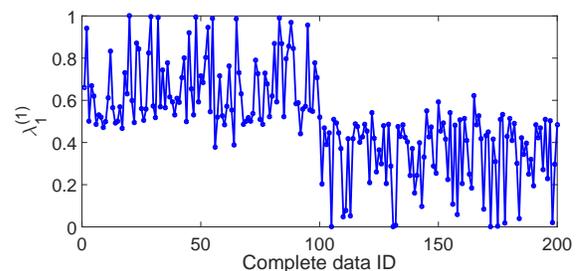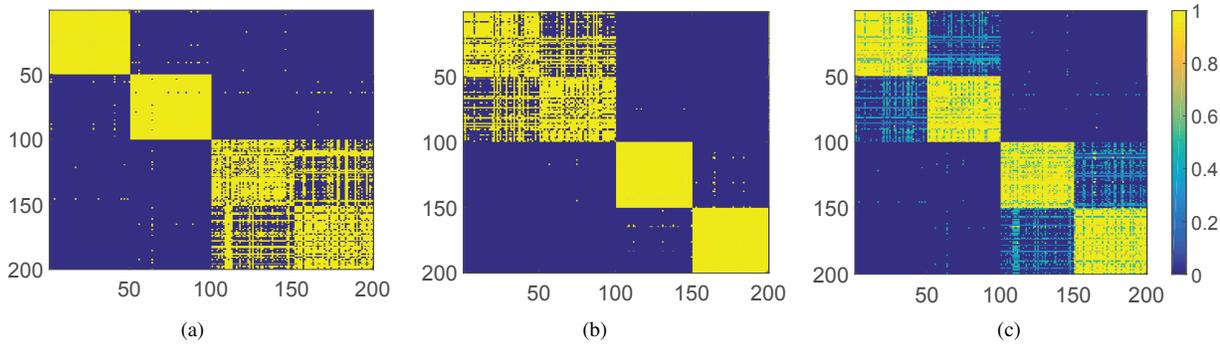The corresponding graphs $G_1^{(1)}$ and $G_1^{(2)}$ constructed using the completed data are illustrated in Fig. 5 (a) and (b). As the first 100 samples were well clustered in the first view, their weights should be greater than those of the other samples. The weights learned by ASGC are shown in Fig. 6. Nearly all the weights $\lambda_1^{(1)}$ of the first 100 samples were greater than 0.5 (i.e., the weights $\lambda_1^{(2)}$ are less than 0.5). This was consistent with the expectation. The combined graph $\mathbb{G}_1$ is shown in Fig. 5 (c), where we can observe that the data with the same label are aligned well. Thus, data of the same shape are grouped together. The new representations learned by Sym-NMF on view1, view2, and ASGC are illustrated in Fig. 7. We can observe that the samples were well clustered using the proposed method. Note that the mean values of $\lambda_1^{(1)}$ and $\lambda_1^{(2)}$ were 0.5016 and 0.4984, respectively, and they express the importance of the first and second views; thus, they can be used as the parameters η_1 and η_2, respectively.

Therefore, the ASGC can be used to combine data from multiple views with different weights for different samples. This demonstrates the effectiveness of the graph combination, and it can be further applied to deal with partial data.

B. Real-world Datasets

The proposed ASGC-PMVC was tested on the following four real-world datasets:

(i) BDGP[1] is a two-view dataset. It contains 2,500 images of drosophila embryos belonging to five categories. Each image is represented by a 1,750-D visual vector and a 79-D text vector for visual and text view, respectively.

(ii) NUS-WIDE[2] is also a two-view dataset selected from Flickr that includes 8,500 samples belonging to five categories (bird, food, sun, tower, and toy). Five hundred visual words and 1,000 tags were used to build the image and text feature spaces, respectively.

(iii) Kitti[3] is an autonomous driving platform containing five driving scenarios (city, residential, road, campus, and person). We selected 10,055 samples with two views: 1,024-D BOW features and 4,096-D VGG deep features.

(iv) COIL-20[4] is the Columbia Object Image Library dataset having 1,440 images belonging to 20 categories. Each category contains 72 images. Three types of features are used to represent the object: 3,304-D LBP, 1,024-D intensity, and 6,750-D Gabor feature spaces.

To simulate the partial view setting, we randomly selected some samples from each view to make the views incomplete. Three criteria were used to describe the missing situation.

(i) Partial view ratio (PVR) is the fraction of partial views of all samples (i.e., PVR=$1 - \sum_{m=1}^{M} \sum_{j=1}^{N} (W^{(m)})_{jj}/(M*N)$).

(ii) Partial sample ratio (PSR) is the fraction of incomplete samples (i.e., PSR=n_2/N).

(iii) PSR of each view (PSRV) is the fraction of partial samples in a given view (i.e., PSRV for the m-th view is $1 - \sum_{j=1}^{N} (W^{(m)})_{jj}/N$).

In the incomplete dataset, when the values of PVR, P-SR, and PSRV were higher, the number of missing samples was greater. When a three-view dataset was considered as an example, there were 600 samples, where 500 samples had incomplete views, and thus PSR=500/600=83%. In the 500 incomplete samples, 200 samples had one missing view, and 300 samples had two missing views, and thus PVR=(200+300*2)/(600*3)=44%. If the first view of the 400 samples was missing, PSRV$_1$=400/600=66%.

We tested three situations to perform a comprehensive test. The first situation considered the most general case in which information loss occurred in a purely random fashion. Different PVR values were simulated in the experiments to test the capability of the proposed framework when dealing with various levels of incompleteness. The second partial-view situation was to randomly select a small fraction of data as complete samples appearing in all views, whereas the remaining data were incomplete samples. To simplify the assignment of incomplete samples to their corresponding views, we evenly distributed them to all views in the experiment according to [18]. For the third situation, we randomly selected some complete samples, and the other views suffered from missing cases. The numbers of incomplete samples in different views were changed. Thus, PSR was fixed, whereas PSRV was varied. Each process was repeated 10 times, and the average

---

[1]http://ranger.uta.edu/%7eheng/Drosophila/
[2]http://lms.comp.nus.edu.sg/research/NUS-WIDE.htm
[3]http://www.cvlibs.net/datasets/kitti/raw_data.php
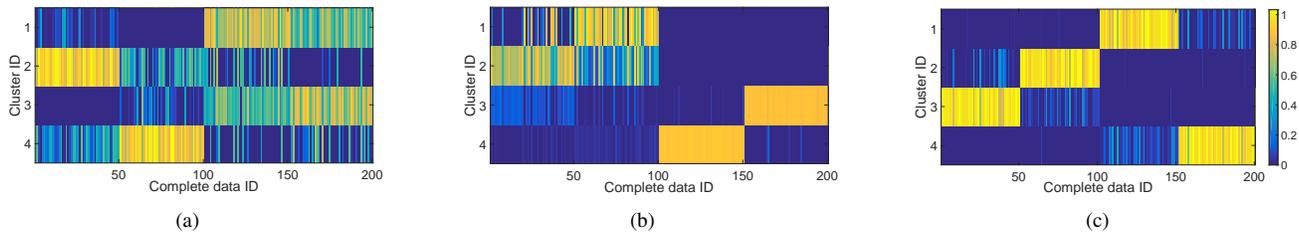[4]http://www.cs.columbia.edu/CAVE/software/softlib/

Fig. 7: New representations learned on (a) $G_1^{(1)}$, (b) $G_1^{(2)}$, (c) $\mathbb{G}_1$.

result was recorded.

C. Methodology

The proposed ASGC-PMVC models were compared with the following baselines.

(i) NMF was performed on partial multiview data. According to [33], the missing data were filled with the average feature values. The coefficient factor was considered as the cluster indicator, and the best result for all views is listed.

(ii) GMultiNMF seeks the common latent subspace from the complete data using the joint NMF model [20], which combines Multiview NMF and graph constraints.

(iii) OMVC is an online multiview clustering algorithm with a joint weighted NMF model that deals with large-scale incomplete views [31].

(iv) PVC establishes the latent subspace with incomplete two-view data [18] but is useful only for two-view data.

(v) GPMVC extends PVC for partial multiview data and introduces the view-specific graph Laplacian regularization in the model [29].

(vi) MVL-IV seeks a common latent subspace by allowing the incomplete data to be restored with the help of complete data [45].

(vii) MIC learns the latent feature representation by integrating the weighted NMF and $L_{2,1}$ regularization [33].

(viii) IMG preserves the compact global structure over the entire heterogeneous data by introducing a graph Laplacian term to couple the incomplete samples [54].

(ix) USL handles incomplete and unlabeled multiview data using a subspace learning framework, which incorporates $L_{2,1}$ and graph Laplacian regularizations [50].

(x) DAIMC is a doubly aligned incomplete multiview clustering algorithm based on weighted semi-NMF with the help of regression [11].

(xi) DIMC projects all data to a common subspace using a deep incomplete multiview clustering model, which incorporates the constraint of the intrinsic geometric structure [56].

Six widely used metrics were adopted to evaluate the clustering performances: accuracy (ACC), normalized mutual information (NMI), adjusted Rand index (AR), F-score, precision, and recall [18], [47], [54], [42], [53], [5]. Each metric illustrates a specific aspect of the clustering result. To compute ACC, each cluster is assigned to the class that is most frequent

in the cluster, and then the accuracy of this assignment is measured by counting the number of correctly assigned samples and dividing by N. NMI determines the amount of statistical information shared by the random variables representing the cluster assignments and user-labeled class assignments of the data points. AR penalizes both false positive and false negative decisions during clustering. F-score supports differential weighting of these two types of errors. Recall measures the fraction of positive examples that are correctly labeled, whereas precision measures that fraction of examples classified as positive that are truly positive. For all these metrics, a higher value indicates better clustering performance. Each metric penalizes or favors different properties in the clustering, and thus we report results on these diverse measures to perform a comprehensive evaluation. These metrics are widely used for evaluating multiview clustering performance with complete or incomplete views. For example, in the compared methods, PVC [18] and USL [50] use NMI to evaluate the clustering performance, and IMG [54] adopts NMI and precision.

D. Adaptive Sample-level Weights

The BDGP dataset was chosen to illustrate the performance of the adaptive sample-level weights learned by the proposed model. BDGP consists of image features and annotation terms. We randomly selected 10% as the complete image-text pairs (i.e., the number of samples in $\mathcal{X}_1$ was 250 (50 samples for each cluster)), and the numbers of partial images and texts were both 1,125. As BDGP is a two-view dataset, we needed to learn only the adaptive weights of complete data. In this subsection, the subscript 1 in X, G, λ, which represents the complete part, is omitted for clear representation.

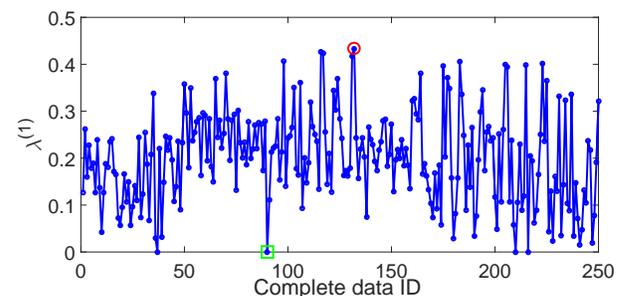Fig. 8: Weight ($\boldsymbol{\lambda}^{(1)}$) of each complete sample in the first view on the BDGP dataset.

Fig. 8 shows the weight ($\boldsymbol{\lambda}^{(1)}$) of each complete sample (X) in the image view. A higher value indicates that the
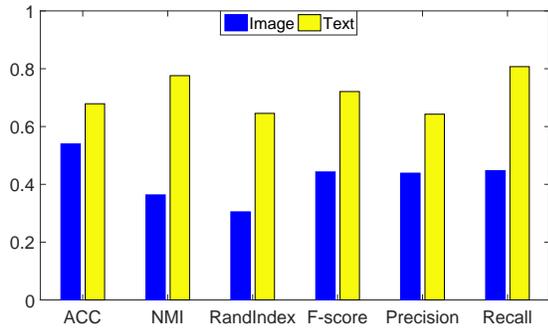
Fig. 9: Clustering results of complete image and text samples on the BDGP dataset

clustering process utilized much more image information than text information for the corresponding sample. It could be observed that different samples contributed differently to the clustering task. Moreover, all the weights of image views were less than 0.5, indicating that the text view ($X^{(2)}$) played a more important role than did image view ($X^{(1)}$). The clustering results on the complete multiview data ($X^{(1)}$) and ($X^{(2)}$) obtained using Sym-NMF [16] are listed in Fig. 9. In addition, it could be observed that the clustering results on the text view were superior to those on the image view. This result further confirms that the learned sample-level weights are reasonable (i.e., $\lambda_i^{(1)} < \lambda_i^{(2)}$ for all samples).
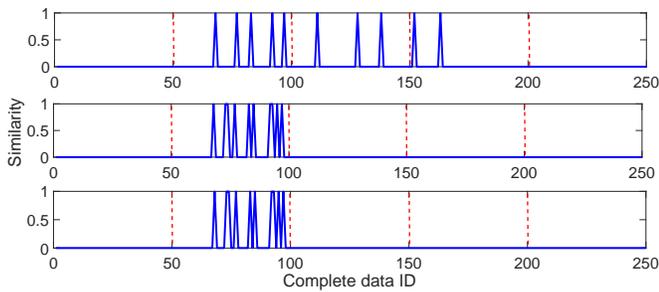
(a) Graph information for the 90-th sample
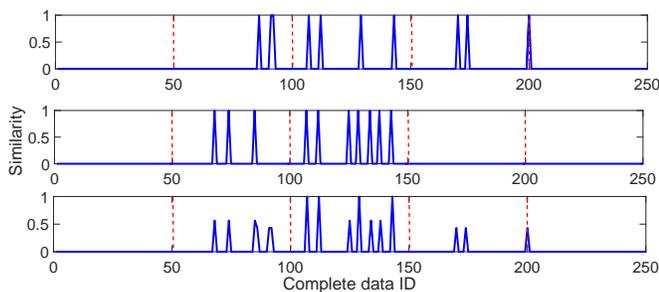
(b) Graph information for the 132-th sample

Fig. 10: Parts of graphs constructed with the complete data on the BDGP dataset, (a) Lines 1–3 are $G_{(90,:)}^{(1)}$, $G_{(90,:)}^{(2)}$, and $\mathbb{G}_{(90,:)}$, respectively. (b) Lines 1–3 are $G_{(132,:)}^{(1)}$, $G_{(132,:)}^{(2)}$, and $\mathbb{G}_{(132,:)}$, respectively.

To investigate the sample-level weights, we checked two typical samples marked by a green square and red circle in Fig. 8. The green square indicates the 90-th complete sample with $\lambda_{90}^{(1)} = 0$ and $\lambda_{90}^{(2)} = 1$, which demonstrates

that the text graph ($G^{(2)}$) is sufficiently clear to illustrate the cluster structure of this sample. Fig. 10(a) shows the corresponding graph information of $G^{(1)}$, $G^{(2)}$, and $\mathbb{G}$ related to the 90-th sample. The red dash lines are used to separate the samples from different clusters. Accordingly, we can conclude that the 90-th sample belongs to the second cluster. It is evident that the samples having higher similarity with the 90-th sample originated from the second cluster according to $G^{(2)}$, whereas they originated from three clusters (Clusters 2–4) according to $G^{(1)}$. The 90-th sample has vague image representation and clear semantic terms. The third line of Fig. 10(a) illustrates the result of $\mathbb{G}_{(90,:)} = \frac{1}{2}\left(\left(\sum_{m=1}^{2} diag(\boldsymbol{\lambda}^{(m)})G^{(m)} + G^{(m)}diag(\boldsymbol{\lambda}^{(m)})\right)\right)_{(90,:)} = G_{(90,:)}^{(2)}$. As $\lambda_{90}^{(1)} = 0$, the second and third lines are identical, indicating that less image information and more text information were used in the learning process.

Similarly, the maximal value of $\boldsymbol{\lambda}^{(1)}$ was 0.43 (i.e., the 132-th sample marked by the red circle in Fig. 8, and thus $\lambda_{132}^{(2)}$ was 1-0.43=0.57. The 132-th sample belongs to Cluster 3, and the neighbors of $G_{(132,:)}^{(1)}$ (shown in the first line of Fig. 10(b)) originated from Clusters 2–4, whereas the neighbors of $G_{(132,:)}^{(2)}$ (shown in the second line of Fig. 10(b)) originated from Clusters 2–3, and most of them are related to the right Cluster 3. Although both image and text views have comparable representations, the text view is clearer than the image view. Thus, $\lambda_{132}^{(1)}$=0.43 and $\lambda_{132}^{(2)}$=0.57 are reasonable. The result of $\mathbb{G}_{(132,:)}$ is illustrated in the third line of Fig. 10(b). It is evident that the 132-th sample has a better similarity to the samples belonging to the third cluster than to the points in the other clusters. As $\boldsymbol{\lambda}^{(m)}$ indicates the importance of each sample in the m-th view, we use its average value to set the parameter $\eta^{(m)}$ in (3) (i.e., $\eta^{(m)} = \frac{1}{n_1}\sum_{i=1}^{n_1}\lambda_i^{(m)}$). In this experiment, $\eta^{(1)} = 0.2021$ and $\eta^{(2)} = 0.7979$.

E. Effect of trade-off parameters
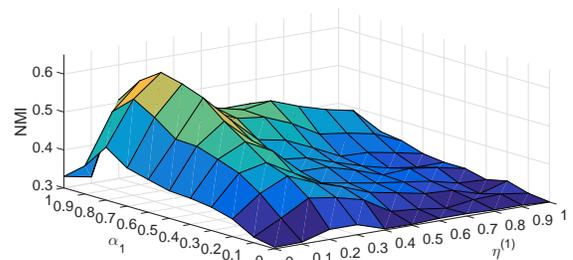
Fig. 11: Effect of different trade-off parameters $\eta^{(1)}$ and α_1 of ASGC-PMVC with PSR 90% on the BDGP dataset.

In the proposed model (3), some parameters exist, namely, trade-off controller $\{\alpha_i\}_{i=1}^2$ and $\{\eta^{(m)}\}_{m=1}^M$. In our experiment, we set $\sum_i \alpha_i = 1$ and $\sum_m \eta^{(m)} = 1$. To test the effects of the parameters, we took the BDGP dataset with PSR 90% as an example, which contained 250 complete pairs and 2,250 incomplete data. We manually adjusted different values of $\eta^{(1)}$ and α_1, where the NMI values are shown in Fig. 11. Of the two parameters, $\eta^{(m)}$ indicates the importance of the

m-th view. We can see that the best clustering performance was obtained when $\eta^{(1)} = 0.2$, which is basically consistent with the value computed by $\boldsymbol{\lambda}^{(m)}$ introduced in the previous subsection ($\eta^{(1)} = 0.2021$). Thus, it verifies that the average value of $\boldsymbol{\lambda}^{(m)}$ could be used as the value of $\eta^{(m)}$ to replace the manual adjustment was verified.

High values of α_1 and α_2 indicate that complete and incomplete data play critical roles in the clustering process, respectively. Fig. 11 reveals that good clustering performance was obtained when the trade-off parameter α_1 was in the range of [0.7, 0.9] and α_2 was in the range of [0.1, 0.3], This confirms that the complete information was more helpful to the clustering task, and it is reasonable to build a bridge between different views for migrating the useful knowledge.

F. Convergence Study

Fig. 12: Convergence curves of the value of the objective function and the corresponding NMI value vs. the number of iterations of ASGC-PMVC with PSR 99% on the BDGP dataset.

We discussed the convergence of the proposed ASGC-PMVC methods in the previous section. In this section, we describe our attempt to numerically evaluate the convergence on the BDGP dataset. The value of the objective function and corresponding NMI value in each iteration are shown in Fig. 12 with PSR 99% for ASGC-PMVC. It can be observed that the value of the objective 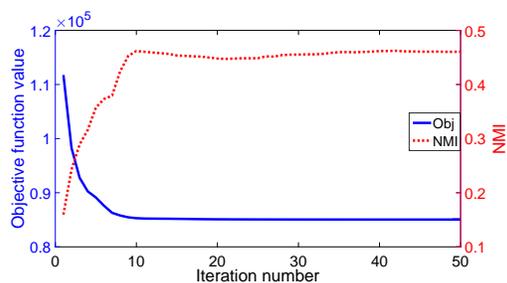function monotonically decreased with the increase in the number of iterations. Moreover, the clustering process converged in a few iterations under this setup, which was sufficient to generate good clustering results.

G. Clustering Results with Different Settings

This subsection details the clustering results under comprehensive experimental setups characterized by different PVRs, PSRs, and PSRVs to account for a wide range of information-loss scenarios in real applications.

1) Clustering results with different PVR values: We first investigated the performance of the proposed method under different PVRs on different datasets and compared the results with the state-of-the-art methods. Note that the PVR value for a two-view dataset could not exceed 50% to ensure each sample had at least one view available in the dataset. Thus, we considered PVRs from 10% to 40% with an interval of 10%.

All results are listed in Table II. BDGP is a balanced dataset with 500 samples in each category. The Kitti dataset contains 10,055 complete samples with both BOW and VGG views.

It is an unbalanced dataset, which made the clustering task more challenging. The numbers of "campus," "city," "person," "resident," and "road" are 1,409, 2,612, 1,768, 2,084, and 2,182, respectively. For the three-view COIL-20 dataset, the number of samples in each cluster is small, which made the clustering task more difficult. Note that PVC and IMG can only be used for two-view data. Therefore, we did not test them on the three-view multiview dataset. The proposed ASGC strategy can make the learned new representation H_1 more precise and is helpful for obtaining the basis factor Z and for representing H_2. As expected, the performance of ASGC-PMVC was better than those of the compared methods.

2) Clustering results with different PSR values: In this experiment, we randomly selected a small fraction of data as complete samples (i.e., they appeared in all views, whereas the others were incomplete samples). As the difference of PSR and PVR cannot be reflected on two-view datasets, we focused on the three-view COIL dataset. Although both PVR and PSRV were fixed at 30%, the PSR was set to range from 50% to 90% with a step of 10%. For a concise presentation, we used NMI as the evaluation criteria in this experiment. Fig. 13 shows the NMI results on the COIL dataset for different testing algorithms when the PSR values were varied. It can be observed that when PSR increased, the performance of all the methods diminished, which indicates the importance of complete samples in terms of learning precise representations.
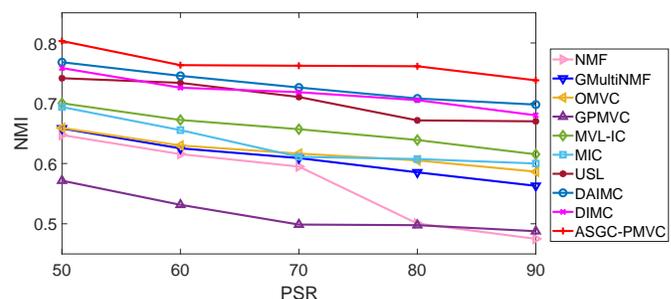
Fig. 13: NMI results for varied PSR values (%) on the COIL dataset.

It could be concluded that most of the multiview learning methods always outperformed the single-view method (NMF), which is consistent with our expectation that multiview data are more informative than single-view data. As GMultiNMF seeks the common latent subspace by considering only a small portion of complete data, it failed to achieve a satisfactory result. The proposed method consistently outperformed all the baselines and achieved promising clustering results. It also benefited from the highlight of informative views of each sample. Note that all the views and samples in the compared methods were treated equally, although different samples in different views may have contributed differently to the clustering process. This may be the primary reason for the higher clustering accuracy achieved by the proposed ASGC-PMVC models.

3) Clustering results with different PSRV values: Finally, we illustrate the clustering performance under different PSRV in each view with a fixed PVR of 30% and PSR of 90%. To be consistent, we also considered a three-view COIL dataset

TABLE II: Clustering results with different PVR values (%)

Dataset	PVR	Methods	ACC	NMI	AR	F-score	Precision	Recall	PVR	ACC	NMI	AR	F-score	Precision	Recall
BDGP	10	NMF	0.7312	0.6077	0.4635	0.5882	0.5052	0.7038	20	0.5701	0.4587	0.1897	0.4081	0.3023	0.6276
		GMultiNMF	0.7612	0.7124	0.6745	0.7470	0.6699	0.8441		0.6808	0.5201	0.5050	0.6147	0.5538	0.6908
		OMVC	0.7712	0.7223	0.6915	0.7634	0.6936	0.8428		0.7061	0.5867	0.5843	0.6306	0.5905	0.7122
		PVC	0.9204	0.7778	0.8091	0.8526	0.8441	0.8523		0.8825	0.7010	0.7202	0.7837	0.7829	0.7821
		GPMVC	0.9203	0.7887	0.8192	0.8485	0.8503	0.8535		0.8946	0.7107	0.7211	0.7926	0.7937	0.7807
		MVL-IC	0.9036	0.7746	0.8040	0.8325	0.8313	0.8343		0.8833	0.7213	0.7235	0.7727	0.8036	0.7926
		MIC	0.8881	0.7602	0.7709	0.8135	0.8162	0.8219		0.8508	0.6992	0.7308	0.7276	0.7355	0.7642
		IMG	0.9205	0.8143	0.8232	0.8426	0.8517	0.8433		0.9004	0.7323	0.7534	0.8103	0.8043	0.8032
		USL	0.8934	0.7937	0.8023	0.8321	0.8334	0.8228		0.8523	0.7047	0.7091	0.7687	0.7798	0.7806
		DAIMC	0.9202	0.8230	0.8342	0.8538	0.8512	0.8504		0.8860	0.7286	0.7817	0.8247	0.8255	0.8135
		DIMC	0.9300	0.8304	0.8432	0.8539	0.8631	0.8541		0.9020	0.7384	0.7788	0.8304	0.8214	0.8141
		ASGC-PMVC	0.9556	0.8727	0.8932	0.9146	0.9132	0.9160		0.9236	0.7889	0.8204	0.8563	0.8560	0.8565
	30	NMF	0.4560	0.3814	0.1140	0.3683	0.2562	0.6762	40	0.4492	0.3302	0.0706	0.3433	0.2336	0.7301
		GMultiNMF	0.5868	0.3440	0.2922	0.4348	0.4309	0.4388		0.4292	0.1502	0.1142	0.3005	0.2853	0.3173
		OMVC	0.6203	0.6033	0.6132	0.6523	0.6639	0.6942		0.6091	0.4788	0.4899	0.5330	0.5706	0.6142
		PVC	0.8368	0.6330	0.6433	0.7448	0.7312	0.7576		0.7836	0.5443	0.5508	0.6807	0.6794	0.6821
		GPMVC	0.8359	0.6440	0.6429	0.7748	0.7373	0.7438		0.7904	0.5642	0.5620	0.6915	0.6919	0.6933
		MVL-IC	0.8340	0.6232	0.6223	0.7640	0.7323	0.7315		0.7730	0.5485	0.5475	0.6739	0.6738	0.6923
		MIC	0.7976	0.6070	0.6078	0.7417	0.7455	0.7664		0.7793	0.5675	0.5642	0.7125	0.6855	0.7047
		IMG	0.8423	0.6395	0.6467	0.7580	0.7503	0.7759		0.8001	0.5932	0.5822	0.7112	0.6837	0.7229
		USL	0.8132	0.6136	0.6229	0.7520	0.7442	0.7526		0.7732	0.5543	0.5630	0.6624	0.6544	0.6628
		DAIMC	0.8326	0.6475	0.6715	0.7358	0.7466	0.7893		0.8029	0.6017	0.6028	0.7023	0.6910	0.7439
		DIMC	0.8242	0.7131	0.7570	0.7921	0.7489	0.7984		0.7857	0.6133	0.6274	0.7472	0.6867	0.7509
		ASGC-PMVC	0.9056	0.7541	0.7807	0.8245	0.8232	0.8259		0.8656	0.6728	0.6964	0.7573	0.7545	0.7600
Kitti	10	NMF	0.4254	0.3299	0.1255	0.3947	0.2814	0.3752	20	0.4064	0.3193	0.1201	0.3867	0.2550	0.3636
		GMultiNMF	0.4288	0.3297	0.1218	0.4059	0.2942	0.4026		0.4208	0.3044	0.0970	0.3832	0.2871	0.3752
		OMVC	0.4455	0.3447	0.1835	0.4154	0.3133	0.4658		0.4203	0.3223	0.1775	0.4036	0.3096	0.4586
		PVC	0.5247	0.3718	0.2437	0.4336	0.3767	0.3973		0.5006	0.3285	0.2188	0.4093	0.3653	0.3837
		GPMVC	0.4961	0.3435	0.2032	0.4366	0.3251	0.4831		0.4832	0.3283	0.1766	0.4110	0.3089	0.4769
		MVL-IC	0.4751	0.2735	0.2453	0.3963	0.3703	0.3703		0.4721	0.2662	0.2214	0.3818	0.3529	0.3729
		MIC	0.5261	0.3637	0.2744	0.4556	0.3820	0.5041		0.4960	0.3597	0.2628	0.4350	0.3822	0.4857
		IMG	0.5134	0.3961	0.3018	0.4732	0.4305	0.5119		0.5113	0.3710	0.2844	0.4594	0.4223	0.4869
		USL	0.5098	0.3897	0.2937	0.4691	0.4358	0.5041		0.5003	0.3839	0.2941	0.4647	0.4128	0.4829
		DAIMC	0.4779	0.3701	0.3211	0.4412	0.4432	0.5277		0.4747	0.3517	0.2937	0.4199	0.4199	0.5084
		DIMC	0.5277	0.4121	0.3062	0.4924	0.4342	0.5036		0.5135	0.3824	0.3039	0.4745	0.4263	0.4976
		ASGC-PMVC	0.5473	0.4183	0.3502	0.5167	0.4566	0.5625		0.5443	0.4039	0.3414	0.5024	0.4523	0.5414
	30	NMF	0.3901	0.2923	0.0844	0.3608	0.2422	0.3439	40	0.3746	0.2890	0.0765	0.3592	0.2405	0.3397
		GMultiNMF	0.3959	0.3028	0.0769	0.3589	0.2611	0.3521		0.3893	0.2944	0.0743	0.3531	0.2524	0.3489
		OMVC	0.4201	0.3006	0.1602	0.3809	0.3002	0.4442		0.4158	0.2917	0.1539	0.3746	0.2974	0.4388
		PVC	0.4768	0.2662	0.1997	0.3711	0.3611	0.3804		0.4601	0.2597	0.1894	0.3682	0.3608	0.3756
		GPMVC	0.4668	0.3107	0.1521	0.3989	0.2847	0.4721		0.4170	0.2921	0.1380	0.3597	0.3073	0.4659
		MVL-IC	0.4711	0.2541	0.1903	0.3623	0.3532	0.3703		0.4380	0.2089	0.1626	0.3437	0.3772	0.3314
		MIC	0.4703	0.3318	0.2408	0.4112	0.3702	0.4601		0.4349	0.2999	0.2144	0.3837	0.3647	0.4480
		IMG	0.5021	0.3702	0.2789	0.4411	0.3921	0.4832		0.4804	0.3655	0.2466	0.3793	0.4024	0.4440
		USL	0.4825	0.3719	0.2651	0.4409	0.3935	0.4723		0.4420	0.3634	0.2691	0.4336	0.3758	0.4515
		DAIMC	0.4498	0.3521	0.2907	0.4321	0.4002	0.4879		0.4190	0.3412	0.2427	0.4310	0.4117	0.4925
		DIMC	0.5136	0.3812	0.3025	0.4511	0.4001	0.4812		0.5108	0.3608	0.2915	0.4345	0.3797	0.4919
		ASGC-PMVC	0.5367	0.3933	0.3219	0.4702	0.4357	0.5221		0.5310	0.3737	0.2927	0.4434	0.4195	0.5129
COIL	10	NMF	0.6024	0.7064	0.3148	0.3133	0.2151	0.5774	20	0.5142	0.6873	0.2531	0.2315	0.1553	0.5366
		GMultiNMF	0.6141	0.7177	0.2869	0.3645	0.2809	0.6259		0.5471	0.6749	0.2603	0.3384	0.2338	0.5439
		OMVC	0.6212	0.7482	0.3185	0.3362	0.3465	0.6916		0.6022	0.7000	0.2360	0.2767	0.2769	0.6198
		GPMVC	0.4666	0.5574	0.4082	0.3129	0.4023	0.4935		0.4297	0.5454	0.3099	0.3106	0.3323	0.3967
		MVL-IC	0.6879	0.7622	0.6357	0.5698	0.6186	0.6409		0.6418	0.7032	0.5627	0.5272	0.5548	0.5878
		MIC	0.7399	0.7152	0.5761	0.5811	0.5776	0.6253		0.6718	0.6925	0.5417	0.5998	0.6143	0.6528
		USL	0.6979	0.7568	0.6641	0.6251	0.5587	0.7405		0.6822	0.7183	0.6057	0.6490	0.5518	0.6699
		DAIMC	0.7005	0.8016	0.6236	0.6418	0.6803	0.6782		0.6150	0.7433		0.6739	0.6483	0.6571
		DIMC	0.7604	0.7625	0.6798	0.6875	0.6532	0.7111		0.6960	0.7373	0.5892	0.6852	0.6091	0.6933
		ASGC-PMVC	0.7595	0.8281	0.7411	0.7315	0.7156	0.7785		0.7219	0.7991	0.6532	0.7220	0.6501	0.7361
	30	NMF	0.5138	0.5949	0.1700	0.2318	0.1538	0.4718	40	0.3572	0.4938	0.0922	0.1592	0.0573	0.2515
		GMultiNMF	0.5414	0.6090	0.1804	0.2414	0.1602	0.4920		0.4525	0.5785	0.2005	0.2392	0.1633	0.3008
		OMVC	0.5501	0.6165	0.2253	0.2558	0.2523	0.5503		0.4700	0.5780	0.2304	0.2683	0.2328	0.4366
		GPMVC	0.4194	0.4987	0.2661	0.3047	0.2861	0.3322		0.4173	0.5009	0.2802	0.2906	0.2910	0.3379
		MVL-IC	0.5635	0.6569	0.5077	0.4898	0.5016	0.5701		0.4941	0.5835	0.3411	0.3623	0.4055	0.4589
		MIC	0.5861	0.6111	0.4982	0.5252	0.4805	0.5791		0.5094	0.5712	0.3574	0.3950	0.4149	0.4643
		USL	0.6119	0.7103	0.5385	0.5603	0.5092	0.6256		0.5245	0.5825	0.3634	0.4068	0.4078	0.4986
		DAIMC	0.5959	0.7261	0.5517	0.5851	0.5523	0.6002		0.5341	0.5895	0.3947	0.4230	0.4155	0.5049
		DIMC	0.6291	0.7183	0.5603	0.5462	0.5290	0.6325		0.5388	0.5678	0.3745	0.4036	0.4235	0.5150
		ASGC-PMVC	0.6722	0.7622	0.6062	0.6273	0.5786	0.6849		0.5789	0.6535	0.4207	0.4691	0.4604	0.5521

and used NMI for evaluation purposes. The PSRV values in the first LBP view (PSRV$_l$) were varied at 10% and 90%. Clustering results are listed in Fig. 14. The LBP feature could obtain a more identifiable structure than intensity and Gabor views, and the clustering performance in the LBP view was better than that in the other two views. Thus, with the loss of numerous LBP features (PSRV$_l$=90%), the performance worsened as compared to the situation with numerous other missing features (PSRV$_l$=10%). We could observe that the proposed ASGC-PMVC model outperformed the compared methods because the adaptive strategy was employed.

4) Clustering results on NUS-WIDE: The NUS-WIDE, which is a naturally incomplete dataset, contains 4,000 complete samples with both visual and text views and 4,500 incomplete samples in which the text information of 1,500 samples and the visual information of 3,000 samples are both missing. Thus, PSR was 53% and PVR was 26%, and of the incomplete data, 18% samples suffered from missing text views, whereas 35% samples suffered from missing image views. Clustering results are listed in Table III, where we can observe that the proposed ASGC-PMVC model outperformed the compared baselines.
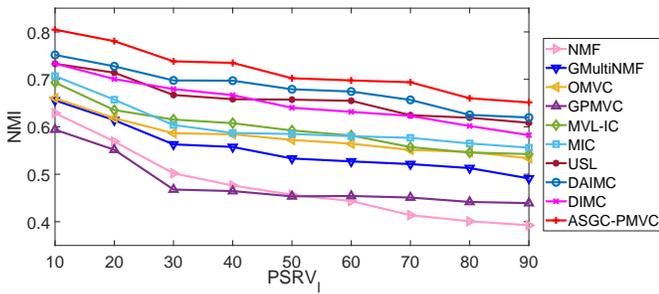
Fig. 14: NMI results for varied PSRV$_l$ values (%) on the COIL dataset.

TABLE III: Clustering results on the NUS-WIDE dataset.

	ACC	NMI	AR	F-score	Precision	Recall
NMF	0.5101	0.3607	0.2130	0.4174	0.3248	0.5924
GmultiNMF	0.5814	0.4009	0.3585	0.4984	0.4569	0.5322
OMVC	0.6805	0.4663	0.4175	0.5523	0.5329	0.5480
PVC	0.6277	0.3216	0.2856	0.4350	0.4156	0.4564
GPMVC	0.6357	0.4281	0.3172	0.4709	0.4355	0.4834
MVL-IC	0.6710	0.4568	0.4073	0.5318	0.4591	0.5301
MIC	0.6922	0.4821	0.4261	0.5631	0.5489	0.5538
IMG	0.7336	0.4875	0.4727	0.5745	0.5693	0.5898
USL	0.7002	<u>0.4901</u>	0.4935	0.5973	<u>0.5916</u>	0.6021
DAIMC	<u>0.7355</u>	0.4756	0.4903	<u>0.6095</u>	0.5852	0.5943
DIMC	0.7251	0.4868	<u>0.4937</u>	0.5926	0.5793	<u>0.6059</u>
ASGC-PMVC	**0.7848**	**0.5152**	**0.5368**	**0.6302**	**0.6244**	**0.6362**

VI. CONCLUSION

In this study, we proposed a novel method called ASGC-PMVC, which uses a small amount of complete data to boost the clustering accuracy on a large amount of partial multiview data. For complete and incomplete data, we considered the different contributions from samples in multiple views to identify the underlying clustering structure. This structure was utilized to guide the clustering process on incomplete multi-view data. Simultaneously, the large amount of incomplete data also provided more information to improve the clustering performance. An efficient iterative algorithm with guaranteed convergence was developed to solve the proposed joint model. Experimental results demonstrated that the proposed model can achieve better performance than the state-of-the-art partial multiview clustering methods.

REFERENCES

[1] S. Bhadra, S. Kaski, and J. Rousu. Multi-view kernel completion. *Machine Learning*, 106(5):713–739, 2017.

[2] S. Bickel and T. Scheffer. Multiview clustering. In *IEEE International Conference on Data Mining*, pages 19–26, 2004.

[3] Y. Cai, Y. Jiao, W. Zhuge, H. Tao, and C. Hou. Partial multi-view spectral clustering. *Neurocomputing*, 311:316–324, 2018.

[4] X. Cao, C. Zhang, C. Zhou, H. Fu, and H. Foroosh. Constrained multi-view video face clustering. *IEEE Transactions on Image Processing*, 24(11):4381–4393, 2015.

[5] M. Cheng, L. Jing, and M. K. Ng. Tensor-based low-dimensional representation learning for multi-view clustering. *IEEE Transactions on Image Processing*, 28(5):2399–2414, 2018.

[6] E. Eaton, M. Desjardins, and S. Jacob. Multi-view constrained clustering with an incomplete mapping between views. *Knowledge and Information Systems*, 38(1):231–257, 2014.

[7] H. Gao, Y. Peng, and S. Jian. Incomplete multi-view clustering. In *International Conference on Intelligent Information Processing*, pages 245–255. Springer, 2016.

[8] X. He, S. Yan, Y. Hu, P. Niyogi, and H. Zhang. Face recognition using Laplacianfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):328–340, 2005.

[9] M. Hong, Z. Luo, and M. Razaviyayn. Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems. *SIAM Journal on Optimization*, 26(1):337–364, 2016.

[10] L. Houthuys and J. Suykens. Unpaired multi-view kernel spectral clustering. In *IEEE Symposium Series on Computational Intelligence*, pages 1–7, 2017.

[11] M. Hu and S. Chen. Doubly aligned incomplete multi-view clustering. In *International Joint Conference on Artificial Intelligence*, pages 2262–2268, 2018.

[12] S. Huang, Z. Kang, I. W. Tsang, and Z. Xu. Auto-weighted multi-view clustering via kernelized graph learning. *Pattern Recognition*, 88:174–184, 2019.

[13] S. Huang, Z. Kang, and Z. Xu. Self-weighted multi-view clustering with soft capped norm. *Knowledge-Based Systems*, 158:1–8, 2018.

[14] C. Kamada, A. Kanezaki, and T. Harada. Probabilistic semi-canonical correlation analysis. In *ACM International Conference on Multimedia*, pages 1131–1134, 2015.

[15] A. Kimura, M. Sugiyama, T. Nakano, H. Hameoka, H. Sakano, E. Maeda, and K. Ishiguro. Semicca: Efficient semi-supervised learning of canonical correlations. *Information and Media Technologies*, 8(2):311–318, 2013.

[16] D. Kuang, C. Ding, and H. Park. Symmetric nonnegative matrix factorization for graph clustering. In *SIAM International Conference on Data Mining*, pages 106–117, 2012.

[17] C. Lampert and O. Kromer. Weakly-paired maximum covariance analysis for multimodal dimensionality reduction and transfer learning. In *European Conference on Computer Vision*, pages 566–579, 2010.

[18] A. Li, Y. Jiang, and Z. Zhou. Partial multi-view clustering. In *Association for the Advancement of Artificial Intelligence*, pages 1968–1974, 2014.

[19] P. Li and S. Chen. Shared gaussian process latent variable model for incomplete multiview clustering. *IEEE Transactions on Cybernetics*, PP(99):1–13, 2018.

[20] J. Liu, C.Wang, J. Gao, and J. Han. Multiview clustering via joint nonnegative matrix factorization. In *SIAM International Conference on Data Mining*, pages 252–260, 2013.

[21] K. Liu, H. Wang, S. Risacher, A. Saykin, and L. Shen. Multiple incomplete views clustering via non-negative matrix factorization with its application in alzheimer's disease analysis. In *International Symposium on Biomedical Imaging*, pages 1402–1405, 2018.

[22] X. Liu, M. Li, L. Wang, Y. Dou, J. Yin, and E. Zhu. Multiple kernel k-means with incomplete kernels. In *Association for the Advancement of Artificial Intelligence*, pages 2259–2265, 2017.

[23] X. Liu, X. Zhu, M. Li, L. Wang, C. Tang, J. Yin, D. Shen, H. Wang, and W. Gao. Late fusion incomplete multi-view clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(99):1–14, 2018.

[24] S. Mehrkanoon and J. Suykens. Regularized semipaired kernel CCA for domain adaptation. *IEEE Transactions on Neural Networks and Learning Systemss*, PP(99):1–15, 2017.

[25] M. K. Ng, Q. Wu, and C. Shen. A fast markov chain based algorithm for miml learning. *Neurocomputing*, 216:763–777, 2016.

[26] F. Nie, G. Cai, J. Li, and X. Li. Auto-weighted multi-view learning for image clustering and semi-supervised classification. *IEEE Transactions on Image Processing*, 27(3):1501–1511, 2018.

[27] F. Nie, J. Li, X. Li, et al. Parameter-free auto-weighted multiple graph learning: A framework for multiview clustering and semi-supervised classification. In *International Joint Conference on Artificial Intelligence*, pages 1881–1887, 2016.

[28] B. Qian, X. Shen, Y. Gu, Z. Tang, and Y. Ding. Double constrained NMF for partial multi-view clustering. In *IEEE International Conference on Digital Image Computing: Techniques and Applications*, pages 1–7, 2016.

[29] N. Rai, S. Negi, S. Chaudhury, and O. Deshmukh. Partial multi-view clustering using graph regularized NMF. In *International Conference on Pattern Recognition*, pages 2192–2197, 2016.

[30] C. Shang, A. Palmer, J. Sun, K.-S. Chen, J. Lu, and J. Bi. Vigan: Missing view imputation with generative adversarial networks. In *IEEE International Conference on Big Data*, pages 766–775, 2017.

[31] W. Shao, L. He, C. Lu, and S. Philip. Online multi-view clustering with incomplete views. In *IEEE International Conference on Big Data*, pages 1012–1017, 2016.

[32] W. Shao, L. He, and S. Philip. Clustering on multi-source incomplete data via tensor modeling and factorization. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 485–497, 2015.

[33] W. Shao, L. He, and P. Yu. Multiple incomplete views clustering via weighted nonnegative matrix factorization with $l_{2,1}$ regularization. In *Jiont European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 318–334, 2015.

[34] W. Shao, X. Shi, and P. Yu. Clustering on multiple incomplete datasets via collective kernel learning. In *IEEE International Conference on Data Mining*, pages 1181–1186, 2013.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2019.2952696, IEEE Transactions on Image Processing

15

[35] X. Shen, F. Shen, and Q. Sun. Semi-paired discrete hashing: Learning latent hash codes for semi-paired cross-view retrieval. *IEEE Transactions on Cybernetics*, PP(99):1–14, 2016.

[36] S. Sun. A survey of multi-view machine learning. *Neural Computing and Applications*, 23(7-8):2031–2038, 2013.

[37] H. Tao, C. Hou, D. Yi, and J. Zhu. Joint embedding learning and low-rank approximation: A framework for incomplete multi-view learning. *arXiv preprint arXiv:1812.10012*, 2018.

[38] H. Tao, C. Hou, D. Yi, and J. Zhu. Unsupervised maximum margin incomplete multi-view clustering. In *International CCF Conference on Artificial Intelligence*, pages 13–25. Springer, 2018.

[39] A. Trivedi, P. Rai, H. D. III, and S. DuVall. Multiview clustering with incomplete views. In *NIPS Workshop on Machine Learning for Social Computing*, pages 1–7, 2010.

[40] Q. Wang, Z. Ding, Z. Tao, Q. Gao, and Y. Fu. Partial multi-view clustering via consistent GAN. In *IEEE International Conference on Data Mining*, pages 1290–1295, 2018.

[41] Y. Wang, W. Yin, and J. Zeng. Global convergence of ADMM in non-convex nonsmooth optimization. In *arXiv preprint arXiv:1511.06324*, pages 1–30, 2015.

[42] J. Wen, Y. Xu, and H. Liu. Incomplete multiview spectral clustering with adaptive graph learning. *IEEE Transactions on Cybernetics*, 2018.

[43] J. Wen, Z. Zhang, Y. Xu, and Z. Zhong. Incomplete multi-view clustering via graph regularized matrix factorization. In *Proceedings of the European Conference on Computer Vision*, pages 1–16, 2018.

[44] J. Wu, W. Zhuge, H. Tao, C. Hou, and Z. Zhang. Incomplete multi-view clustering via structured graph learning. In *Pacific Rim International Conference on Artificial Intelligence*, pages 98–112, 2018.

[45] C. Xu, D. Tao, and C. Xu. Multi-view learning with incomplete views. *IEEE Transactions on Image Processing*, 24(12):5812–5825, 2015.

[46] J. Xu, J. Han, F. Nie, and X. Li. Re-weighted discriminatively embedded k-means for multi-view clustering. *IEEE Transactions on Image Processing*, 26(6):3016–3027, 2017.

[47] N. Xu, Y. Guo, X. Zheng, Q. Wang, and X. Luo. Partial multi-view subspace clustering. In *ACM Multimedia Conference on Multimedia Conference*, pages 1794–1801, 2018.

[48] W. Yang, Y. Shi, Y. Gao, L. Wang, and M. Yang. Incomplete-data oriented multiview dimension reduction via sparse low-rank representation. *IEEE Transactions on Neural Networks and Learning Systems*, PP(99):1–16, 2018.

[49] Y. Ye, X. Liu, Q. Liu, and J. Yin. Consensus kernel k-means clustering for incomplete multi-view ddata. *Computational Intelligence and Neuroscience*, pages 1–11, 2017.

[50] Q. Yin, S. Wu, and L. Wang. Unified subspace learning for incomplete and unlabeled multi-view data. *Pattern Recognition*, 67:313–327, 2017.

[51] Q. Yin, S. Wu, and L. Wang. Multiview clustering via unified and view-specific embeddings learning. *IEEE Transactions on Neural Networks and Learning Systems*, PP(99):1–13, 2018.

[52] B. Zhang, J. Hao, G. Ma, J. Yue, and Z. Shi. Semi-paired probabilistic canonical correlation analysis. In *International Conference on Intelligent Information Processing*, pages 1–10, 2014.

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

[54] H. Zhao, H. Liu, and Y. Fu. Incomplete multi-modal visual data grouping. In *International Joint Conference on Artificial Intelligence*, pages 2392–2398, 2016.

[55] J. Zhao, X. Xie, X. Xu, and S. Sun. Multi-view learning overview: Recent progress and new challenges. *Information Fusion*, 38:43–54, 2017.

[56] L. Zhao, Z. Chen, Y. Yang, Z. J. Wang, and V. C. Leung. Incomplete multi-view clustering via deep semantic mapping. *Neurocomputing*, 275:1053–1062, 2018.

[57] X. Zhou, X. Chen, and S. Chen. Neighborhood correlation analysis for semi-paired two-view data. *Neural Processing Letters*, 37(3):335–354, 2013.

[58] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision*, pages 2223–2232, 2017.

[59] X. Zhu, X. Liu, M. Li, E. Zhu, L. Liu, Z. Cai, J. Yin, and W. Gao. Localized incomplete multiple kernel k-means. In *International Joint Conference on Artificial Intelligence*, pages 3271–3277, 2018.

[60] L. Zong, X. Zhang, and X. Liu. Multi-view clustering on unmapped data via constrained non-negative matrix factorization. *Neural Networks*, 108:155–171, 2018.

Liu Yang received the Ph.D. degree in computer science from the School of Computer and Information Technology, Beijing Jiaotong University, China, in 2016. She is currently an Associate Professor with the College of Intelligence and Computing, Tianjin University. Her research interests include data mining and machine learning.

Chenyang Shen received the B.Sc. degree from Yangzhou University in 2010, and the M.Phil. and Ph.D. degrees from Hong Kong Baptist University in 2012 and 2015, respectively. He is an Instructor with the Division of Medical Physics and Engineering, Department of Radiation Oncology, University of Texas Southwestern Medical Center. His current research interests include medical imaging, scientific computing, data mining, and deep learning.

Qinghua Hu received the B.S., M.S. and Ph.D degrees from the Harbin Institute of Technology, Harbin, China, in 1999, 2002, and 2008, respectively. He was a Post-Doctoral Fellow with the Department of Computing, Hong Kong Polytechnic University, from 2009 to 2011. He is currently the Dean of the School of Artificial Intelligence, the Vice Chairman of the Tianjin Branch of China Computer Federation, and the Vice Director of the SIG Granular Computing and Knowledge Discovery. He is currently supported by the Key Program, National Natural Science Foundation of China. He has published over 200 peer-reviewed papers. His current research is focused on uncertainty modeling in big data, machine learning with multi-modality data, and intelligent unmanned systems. He is an Associate Editor of the IEEE Transactions on Fuzzy Systems, Acta Automatica Sinica, and Energies.

Liping Jing Liping Jing is a Professor and the Dean of Computer Science Department in Beijing Jiaotong University. She received the Ph.D. degree in applied mathematics from the University of Hong Kong, in 2007. Her research is focused on Machine Learning, High-dimensional data representation and their application in the area of Artificial Intelligent. Now she is supported by Outstanding Youth Science Foundation, National Natural Science Foundation of China, Key Program, Beijing Natural Science Foundation. She has published more than 90 peer-reviewed papers.

Yingbo Li received his Master degree in Statistics from University of Pittsburgh, USA, in 2006. He has worked on Data Analysis for over 10 years. Currently he is working for China Automotive Technology and Research Center as Senior Data Engineer.