Fast Incomplete Multi-View Clustering With View-Independent Anchors

Preprint *in* IEEE Transactions on Neural Networks and Learning Systems · May 2022

DOI: 10.1109/TNNLS.2022.3220486

4 authors:

Suyuan Liu
National University of Defense Technology
5 PUBLICATIONS **17** CITATIONS
SEE PROFILE

Xinwang Liu
National University of Defense Technology
238 PUBLICATIONS **7,118** CITATIONS
SEE PROFILE

Siwei Wang
National University of Defense Technology
54 PUBLICATIONS **556** CITATIONS
SEE PROFILE

En Zhu
National University of Defense Technology
195 PUBLICATIONS **3,030** CITATIONS
SEE PROFILE

Fast Incomplete Multi-view Clustering with View-independent Anchors

Suyuan Liu, Xinwang Liu*, *Senior Member, IEEE*, Siwei Wang, Xin Niu, and En Zhu

***Abstract*—Multi-view clustering methods aim to exploit consistent and complementary information among each view and achieve encouraging performance improvement than single-view counterparts. In practical applications, it is common to obtain instances with partially-available information, raising researches of incomplete multi-view clustering (IMC) issues. Recently, several fast IMC methods have been proposed to process the large-scale partial data. Though with considerable acceleration, these methods seek view-shared anchors and ignore specific information among single views. To tackle the above issue, we propose a fast incomplete multi-view clustering with view-independent anchors (FIMVC-VIA) method in this paper. Specifically, we learn individual anchors based on the diversity of distribution among each incomplete view and construct a unified anchor graph following the principle of consistent clustering structure. By constructing an anchor graph instead of pair-wise full graph, the time and space complexities of our proposed FIMVC-VIA are proven to be linearly related to the number of samples, which can efficiently solve the large-scale task. The experiment performed on benchmarks with different missing rate illustrates the improvement in complexity and effectiveness of our method compared to other IMC methods. Our code is publicly available at https://github.com/Tracesource/FIMVC-VIA.**

***Index Terms*—Multi-view Clustering, Incomplete Multi-view Clustering, Large-scale Clustering, Anchor Graph**

I. Introduction

In the real world, data in many domains can be represented with various features from different views, which is known as multi-view data in community [1]–[4]. For example, the same piece of news can be presented in multiple formats, such as text, audio and video [5]. Integrating the provided diverse information for clustering has turned out to be a critical task in unsupervised learning, and many multi-view clustering (MVC) approaches have been presented [6]–[9]. For instance, co-training MVC methods try to explore the consistency among multiple views by iteratively exchanging the information learned on each view [10]–[12]. Multi-kernel clustering methods aim to learn a common kernel or consistent representation based on the predefined kernels, and finally obtain a unified clustering results [13], [14]. According to the consensus latent subspace consumption, multi-view subspace methods propose to get a common features among all views [15]–[17]. The vital part of MVC is to learn a unified representation based on exploring consistency among different views. Most of the existing MVC work assume that instances on all views are available [18]–[21].

S. Liu, X. Liu, S. Wang, X. Niu and E. Zhu are with School of Computer, National University of Defense Technology, Changsha, China, 410073 (e-mail: {suyuanliu, wangsiwei13, xinwangliu, niuxin, enzhu}@nudt.edu.cn.)

* Corresponding author.

However, in most practical applications, samples are often lacking information on some views [22]. For example, the news may be reported in multiple views, such as text, audio, and video, but not all the news appears in all views. In social network analysis, it is impossible for people to sign up for all social platforms, which causes incompleteness of the instances in the respective views. The unpaired missing views make it hard to learn the common representation among all views. Therefore, traditional data integrity-based approaches have difficulty with incomplete multi-view data [23], [24].

To handle the above problem, several incomplete multi-view clustering (IMC) methods have been proposed [25]–[27]. The existing IMC approaches can be concluded with the following three main strategies: matrix factorization, graph or kernel construction and deep learning. As the first IMC method using matrix factorization, PVC learns a consensus latent representation from incomplete data with NMF and ℓ_1 constraint [28]. Based on PVC, [29] propose to capture the global structure with manifold learning. MIC integrates joint weighted NMF and $\ell_{2,1}$ regularization for better performance [30]. The IMC methods with graph or kernel construction attempt to generate a common similarity matrix among different views. [31] jointly fill in the absent value and learn a clustering representation. Meanwhile, UEAF units incomplete view inferring and common embedding generation into a unified framework [32]. Deep IMC methods infer missing data and extract representation with a deep neural network [33], [34]. As a representative, Dimc-net extracts the high-level information of each view and learns a common representation [35]. Furthermore, [36] integrates feature extracting and data recovery jointly by applying contrastive learning. However, many existing methods have problems with handling data of three or more views, which makes it challenging to tackle complex absent data in real-world applications. Moreover, the quadratic or even cubic complexities of time and space restrict the availability of existing algorithms on large-scale datasets [37].

There have been many effective solutions to the high complexity problem in complete multi-view clustering. Among these methods, anchor-based methods have recently gained a lot of attention due to their efficiency and scalability [38], [39]. For example, [40] propose to first get the anchor points by k-means, then construct the anchor graphs on each view, and finally collocate them to get a unified representation. The handling of large-scale missing data remains a challenge on IMC. [24] and [41] propose to solve the large-scale IMC problems with consistent learned anchors, which effectively reduces the complexity to be linear to samples. However,

this approach assumes that all views share the same anchor points, ignoring geometric structures specific to each view. In addition, although the inclusion of projection matrices solves the problem of inconsistent dimensionality of uniform anchor points across views, it also introduces more time and space consumption.

In this paper, we propose Fast Incomplete Multi-view Clustering with View-independent Anchors (FIMVC-VIA) to handle the above issues. Specifically, we learn view-independent anchor matrices for exploring complementary and construct a consensus anchor graph to keep consistency among views together. The space complexity of the constructed graph is significantly reduced from $\mathcal{O}\left(n^2\right)$ to $\mathcal{O}\left(nm\right)$ with an anchor graph instead of a full graph to measure the similarity. Furthermore, we balance the impact of each view with an adaptive weighting coefficient. To address the consequential optimization problem, a three-step alternative optimization algorithm with guaranteed convergence is designed in this paper. Extensive experimental results can validate that our proposed method performs effectively and efficiently on seven benchmark datasets. In summary, the proposed FIMVC-VIA has the following contributions,

- We propose a flexible and scalable IMC method termed as FIMVC-VIA to solve large-scale IMC problems. With time and space complexity linear to the number of samples, our proposed method can effectively handle large-scale data.
- FIMVC-VIA learns individual anchors on each view for the complementary information and constructs a consensus anchor graph to keep consistence among each view.
- Experimental results conducted on multiple partial datasets prove the superiority of our proposed FIMVC-VIA over SOTA IMC approaches.

The remaining sections of our paper are organized as follows. Section II outlines some relevant works of FIMVC-VIA. Section III proposes our FIMVC-VIA approach and solves it with a three-step alternative algorithm. Section IV evaluates our experiments. Section V is the summary.

II. Related Work

A summary of multi-view clustering and incomplete multi-view clustering is provided in this section, as well as a detailed introduction to graph-based IMC method.

First, we give the definitions of the main notations used in this paper. The letters in bold upper-case (e.g., $\mathbf{X}$), bold lower-case (e.g., $\boldsymbol{\gamma}$), and lower-case (e.g., n) denote matrices, vectors, and scalars, respectively. For a matrix $\mathbf{U} \in \mathbb{R}^{n_1 \times n_2}$, we use $\mathbf{u}_j$ and $u_{i,j}$ to denote the j-th column and the ij-th entry of $\mathbf{U}$. $\|\mathbf{U}\|_F$ and $\mathrm{Tr}\left(\mathbf{U}\right)$ indicate the Frobenius norm and trace of $\mathbf{U}$. Further, $\mathbf{1}$ denotes a vector with all values of one and $\mathbf{I}_m$ denotes an identity matrix of size $m \times m$.

A. Multi-view Clustering

Multi-view clustering has attracted much attention because of its remarkable effectiveness in unsupervised tasks. The goal of MVC is to exploit consistent and complementary information on each view to get a merged result. We conclude the existing MVC approaches into four types: co-training clustering, multi-kernel clustering, subspace clustering and graph-based clustering.

Co-training MVC methods execute the strategy of clustering with knowledge iteratively learning from different views to maximize the consistency among views [42], [43]. MVSim [44] first implements the co-training algorithms into the MVC tasks to process all the data of each view simultaneously. Following MVSim, [45] propose to use sparse matrix decomposition to group subjects consistently.

Multi-kernel clustering methods derive a unified kernel by combining predefined kernels on each view in different ways [46]–[49]. For example, MKKM [50] optimizes the weights of each kernel and the consensus clustering representation jointly. Based on MKKM, [51] learns the optimally combined coefficients by introducing matrix-induced regularization terms to generate a unified kernel with less redundant information.

Multi-view subspace methods attempt to learn a consistent representation among all views with latent subspace [40], [52]. [53] attempt an accurate and robust representation with reconstructed data led in the latent subspace. [54] further explore the consistency and specificity of the subspace representation jointly.

The above methods are effective in solving the MVC problem, but they do not take full advantage of the structure information. To capture the structure information, multi-view graph clustering methods try to seek a fused graph. In this category, [55] develop an adaptively weighted graph clustering method to increase the robustness. To avoid the issue that previous methods must perform post-processing to obtain clustering results and fall into local optimal solutions, MVGL [56] outputs the unified graph with exactly k-connected components to obtain the clustering indicators directly. The performance of the aforementioned methods significantly relies on the initial graphs. MLAN [57] solves it by learning the local structure and performing clustering simultaneously.

B. Incomplete Multi-view Clustering

In real applications, the data to be clustered is always partially missing on some views, which leads to the so-called incomplete multi-view clustering (IMC) problem. However, most of the previous MVC methods are conducted with the assumption that all samples were complete in each view. Consequently, clustering with partial data has become a challenging and worthwhile issue. Similar to the MVC approaches, we can divide the existing IMC approaches into three types: matrix factorization approaches, graph or kernel construction approaches, and deep learning approaches. Matrix factorization IMC approaches learn a consensus representation from incomplete data with view-specific coefficients. For example, [58] tries to get a consistent basis matrix with view-specific weight matrix. In [32], an error matrix was introduced to recover the incomplete views. [59] learns view-specific latent representations and seeks a shared clustering representation jointly. Graph or kernel construction IMC approaches are devoted to reconstructing a unified graph or kernel matrix between multiple views. [60] proposes to pad the similarity

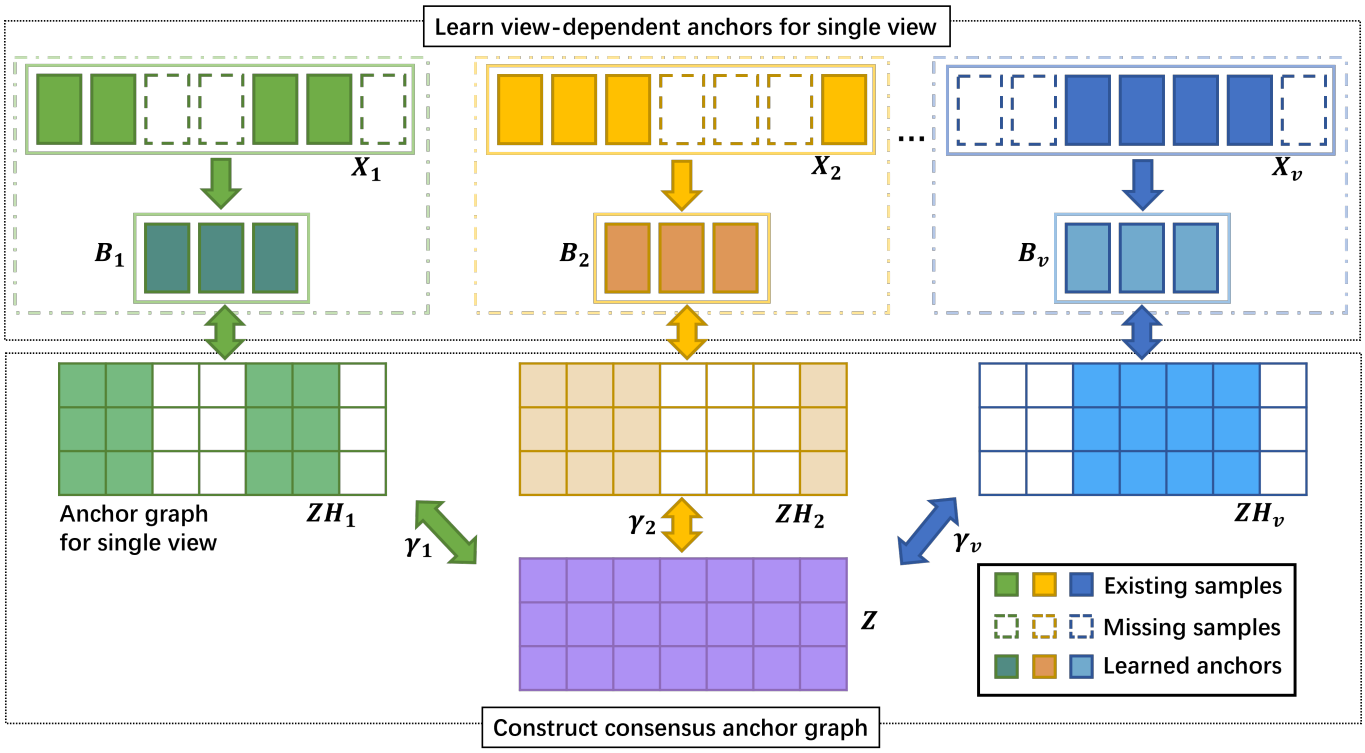


Fig. 1: The framework of FIMVC-VIA. In each iteration, FIMVC-VIA first learns the independent anchor points on each view. Then the weights on each view are learned based on the learned anchor points. Finally, we construct the consensus anchor graph with view-independent anchors and coefficients.

matrix by spectral perturbation theory. Inspired by genetics, [61] explores the consistent part and unique par among incomplete views. [62] proposes to use samples present in all views as anchors and perform spectral clustering to get the final result. With the rapid development of deep learning [63]–[65], deep learning IMC approaches have achieved good performance by extracting high-level information via deep neural networks. Among these methods, graph construction methods have received a lot of attention due to their advantages in capturing data structures.

Given multi-view data $\mathbf{X}_p \in \mathbb{R}^{d_p \times n}$ in the p-th view, we utilize the indicator matrix $\mathbf{H}_p \in \{0,1\}^{n \times n_p}$ to show the missing parts of the data, where n_p is the number of existing samples in the p-th view. Assume there is a given indicator vector $\mathbf{w}_p \in \mathbb{R}^{n_p}$ that contains the index for n_p existing samples for the p-th view in order, then $\mathbf{H}_{p,j,l} = 1$ when $\mathbf{w}_{p,j} = l$. With the defined indicator matrix, $\mathbf{X}_p\mathbf{H}_p \in \mathbb{R}^{d_p \times n_p}$ is the ordered complete matrix of all the data present in the p-th view.

Because of partial absent instances, the full graph may drop some empty blanks in the respective rows and columns. Then, by placing specific sub-graphs into respective parts, it optimizes a merged full graph. The general framework of graph-based IMC can be mathematically represented in two stages:

$$\min_{\mathbf{S}_p} \|\mathbf{X}_p\mathbf{H}_p - \mathbf{X}_p\mathbf{H}_p\mathbf{S}_p\|_{\mathbf{F}}^2 + f(\mathbf{S}_p), s.t. \mathbf{S}_p \geq 0, {\mathbf{S}_p}^\top \mathbf{1} = \mathbf{1}, \tag{1}$$

where $\mathbf{S}_p \in \mathbb{R}^{n_p \times n_p}$ denotes the generated corresponding sub-graph in the p-th view, and $f(\cdot)$ represents the regularization term. After optimizing to obtain the sub-graph $\mathbf{S}_p$ for each view, the respective complete graph can be reconstructed with $\mathbf{H}_p\mathbf{S}_p{\mathbf{H}_p}^\top$. The second stage is to fuse the graphs recovered from each view to obtain a consensus graph $\mathbf{S}$. Finally, spectral clustering algorithm is performed on the consistent $\mathbf{S}$.

Based on this scheme, many approaches have enhanced the clustering performance by employing different regularization terms [47]. However, many previous methods cannot handle data with over two views, which limits their application scenarios [66], [67]. Moreover, most of the existing approaches are limited by high time and space complexity and cannot cope with large-scale tasks. We will propose FIMVC-VIA for applying on large-scale IMC tasks in the next section.

III. Proposed Method

This section starts with the formulation, then moves on to the optimization procedure, and lastly to the convergence and complexity analysis.

A. Formulation

The existing graph-based IMC methods involve the construction of full graphs, which makes them suffer from the time complexity of $\mathcal{O}\left(n^3\right)$ and the space complexity of $\mathcal{O}\left(n^2\right)$. In order to tackle large-scale IMC problems, we decide to construct anchor graphs with m selected anchor points instead of full graphs. The existing methods typically select the anchors with k-means or sampling strategy, which seriously affects the anchor quality on partial data [24]. To address

the above problems, we propose to learn rather than select anchors based on the presence samples. Moreover, we learn anchors on each view individually to explore complementarity between views. Defining the anchor matrix in the p-th view as $\mathbf{B}_p \in \mathbb{R}^{d_p \times m}$ where m is the amount of anchors, the anchor graph $\mathbf{Z}_p \in \mathbb{R}^{m \times n}$ can be constructed as follows,

$$\min_{\mathbf{Z}_p} \|\mathbf{X}_p\mathbf{H}_p - \mathbf{B}_p\mathbf{Z}_p\mathbf{H}_p\|_{\mathbf{F}}^2 + f(\mathbf{Z}_p), s.t. \mathbf{Z}_p \geq 0, \mathbf{Z}_p^\top \mathbf{1} = \mathbf{1}, \tag{2}$$

where $\mathbf{X}_p\mathbf{H}_p$ represents the available instances in the p-th view.

To follow the principle that all views in MVC should share the same graph structure, we decide to construct a consistent anchor graph. In addition, we impose an orthogonal constraint on each anchor matrix to enhance its discriminative property. The strategy of learning unified anchors is under the assumption that the common features are shared across views, while the strategy of learning view-specific anchors assumes that the structure of the data on each view is consistent. Unlike IMVC-CBG [24], instead of using a projection matrix to project the unified anchor points on each view, we learn an independent set of anchor points directly on each view. With the view-specific anchor matrix $\mathbf{B}_p \in \mathbb{R}^{d_p \times m}$ and the consistent anchor graph $\mathbf{Z} \in \mathbb{R}^{m \times n}$, we define the optimization for FIMVC-VIA as follows,

$$\begin{aligned} \min_{\boldsymbol{\gamma}, \{\mathbf{B}_p\}_{p=1}^v, \mathbf{Z}} & \sum_{p=1}^{v} \gamma_p^2 \|\mathbf{X}_p\mathbf{H}_p - \mathbf{B}_p\mathbf{Z}\mathbf{H}_p\|_{\mathbf{F}}^2 + \mu \|\mathbf{Z}\|_{\mathbf{F}}^2, \\ & s.t. \boldsymbol{\gamma}^\top \mathbf{1} = 1, \mathbf{B}_p^\top \mathbf{B}_p = \mathbf{I}_m, \mathbf{Z} \geq 0, \mathbf{Z}^\top \mathbf{1} = \mathbf{1}, \end{aligned} \tag{3}$$

where $\mathbf{H}_p \in \{0,1\}^{n \times n_p}$ is the indicator matrix for marking unavailable data, and μ is the hyper-parameter balancing the impact between consistent anchor graph construction and regularization term.

B. Optimization

When considering all variables jointly, optimizing Eq. (3) yields a non-convex problem, and we solve it with an alternating optimization algorithm. To optimize the process in a simplified manner, we observe that $\mathbf{X}_p\mathbf{H}_p{\mathbf{H}_p}^\top = \mathbf{X}_p \otimes \mathbf{A}_p$ where $\mathbf{A}_p = \mathbf{1}_{d_p}\mathbf{h}_p$ and $\mathbf{h}_p = [h_{p,1}, ..., h_{p,n}]^\top$ with $h_{p,j} = \sum_{l=1}^{n_p} \mathbf{H}_{p,l,j}$, $\otimes$ denotes the Hadamard product. The space complexity is reduced from $\mathcal{O}(vn^2)$ to $\mathcal{O}(dn)$ with such a transformation.

1) Optimize anchor matrix $\{\mathbf{B}_p\}_{p=1}^v$: When $\boldsymbol{\gamma}$ and $\mathbf{Z}$ are fixed, the optimization for $\{\mathbf{B}_p\}_{p=1}^v$ can be written as follows,

$$\min_{\{\mathbf{B}_p\}_{p=1}^v} \sum_{p=1}^{v} \gamma_p^2 \|\mathbf{X}_p\mathbf{H}_p - \mathbf{B}_p\mathbf{Z}\mathbf{H}_p\|_{\mathbf{F}}^2, s.t. \mathbf{B}_p^\top \mathbf{B}_p = \mathbf{I}_m. \tag{4}$$

Noticed that each $\mathbf{B}_p$ is uncorrelated when optimizing, we transform Eq. (4) into the following form by removing irrelevant items,

$$\max_{\mathbf{B}_p} \operatorname{Tr}(\mathbf{B}_p\boldsymbol{\Lambda}_p), s.t. \mathbf{B}_p^\top \mathbf{B}_p = \mathbf{I}_m, \tag{5}$$

where $\boldsymbol{\Lambda}_p = (\mathbf{X}_p \otimes \mathbf{A}_p)\mathbf{Z}^\top$. By performing a singular value decomposition on $\boldsymbol{\Lambda}_p$, we can derive the optimal solution for $\mathbf{B}_p$ as $\boldsymbol{\Sigma}_m\boldsymbol{\Psi}_m^\top$, where $\boldsymbol{\Sigma}_m$ and $\boldsymbol{\Psi}_m$ denotes the matrices consist of the first m left singular vectors and the first m right singular vectors of $\boldsymbol{\Lambda}_p$. It cost the time complexity of $\mathcal{O}(nmd + m^2d)$ to obtain all optimal $\{\mathbf{B}_p\}_{p=1}^v$, where $d = \sum_{i=1}^v d_p$.

2) Optimize consistent anchor graph $\mathbf{Z}$: When $\{\mathbf{B}_p\}_{p=1}^v$ and $\boldsymbol{\gamma}$ are fixed, optimizing $\mathbf{Z}$ equals to solve the following problem:

$$\begin{aligned} \min_{\mathbf{Z}} & \sum_{p=1}^{v} \gamma_p^2 \|\mathbf{X}_p\mathbf{H}_p - \mathbf{B}_p\mathbf{Z}\mathbf{H}_p\|_{\mathbf{F}}^2 + \mu \|\mathbf{Z}\|_{\mathbf{F}}^2, \\ & s.t. \mathbf{Z} \geq 0, \mathbf{Z}^\top \mathbf{1} = \mathbf{1}, \end{aligned} \tag{6}$$

By removing the irrelevant items, we rewrite Eq. (6) as follows:

$$\begin{aligned} \min_{\mathbf{Z}} & \sum_{i=1}^{v} \gamma_p^2 \operatorname{Tr}\left(\mathbf{Z}^\top\mathbf{Z}\left(\mathbf{Q}_p + \frac{\mu}{\gamma_p^2}\mathbf{I}_n\right) - 2{\mathbf{X}_p}^\top\mathbf{B}_p\mathbf{Z}\mathbf{Q}_p\right), \\ & s.t. \mathbf{Z} \geq 0, \mathbf{Z}^\top \mathbf{1} = \mathbf{1}, \end{aligned} \tag{7}$$

where $\mathbf{Q}_p = \mathbf{H}_p{\mathbf{H}_p}^\top$. Then with $\mathbf{X}_p\mathbf{H}_p{\mathbf{H}_p}^\top = \mathbf{X}_p \otimes \mathbf{A}_p$, Eq. (7) is equal to:

$$\begin{aligned} \min_{\mathbf{Z}} & \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{p=1}^{v} \gamma_p^2 \left(\left(\mathbf{A}_{p,1,j} + \frac{\mu}{\gamma_p^2}\right)\mathbf{Z}_{j,i}^2 - 2\mathbf{A}_{p,1,j}\mathbf{C}_{p,i,j}\mathbf{Z}_{j,i}\right), \\ & s.t. \mathbf{Z} \geq 0, \mathbf{Z}^\top \mathbf{1} = \mathbf{1}, \end{aligned} \tag{8}$$

where $\mathbf{C}_p = {\mathbf{X}_p}^\top\mathbf{B}_p$. Denoting $\mathbf{z}_i$ as a vector with $z_{j,i}$ to be the j-th element, Eq. (8) can solve by column as the following form,

$$\min_{\mathbf{z}_i} \|\mathbf{z}_i - \mathbf{y}_i\|_{\mathbf{F}}^2, s.t. \mathbf{z}_i \geq 0, \mathbf{z}_i^\top \mathbf{1} = 1, \tag{9}$$

where $\mathbf{y}_i^\top = \frac{\sum_{p=1}^v \gamma_p^2 \mathbf{A}_{p,1,j}\mathbf{X}_{p,:,i}^\top\mathbf{B}_p}{\mu + \sum_{p=1}^v \gamma_p^2 \mathbf{A}_{p,1,j}}$. We can solve the above problem efficiently. First, we write the Lagrangian function of Eq. (9) as follows,

$$\mathcal{L}(\mathbf{z}_i, \sigma_i, \alpha_i) = \|\mathbf{z}_i - \mathbf{y}_i\|_{\mathbf{F}}^2 - \sigma_i\left(\mathbf{z}_i^\top\mathbf{1} - 1\right) - {\alpha_i}^\top\mathbf{z}_i, \tag{10}$$

where σ_i and α_i are the corresponding Lagrangian multipliers. We have the KKT conditions as,

$$\begin{cases} \mathbf{z}_i - \mathbf{y}_i - \sigma_i\mathbf{1} - \alpha_i = 0, \\ \alpha_i \otimes \mathbf{z}_i = 0. \end{cases} \tag{11}$$

Together with the constraint $\mathbf{z}_i^\top\mathbf{1} = 1$, We have the following equation:

$$\mathbf{z}_i = \max(\mathbf{y}_i + \sigma_i\mathbf{1}, 0), \sigma_i = \frac{1 + {\mathbf{y}_i}^\top\mathbf{1}}{m}. \tag{12}$$

The time complexity of optimizing $\mathbf{Z}$ is $\mathcal{O}(nmd)$ with the close-form solution.

3) Optimize view weight γ: When the irrelevant variables are fixed, the optimization problem w.r.t. $\boldsymbol{\gamma}$ can be formulated as

$$\min_{\boldsymbol{\gamma}} \sum_{p=1}^{v} {\gamma_p}^2 \epsilon_p, s.t. \boldsymbol{\gamma}^\top \mathbf{1} = 1, \boldsymbol{\gamma} \geq 0, \tag{13}$$

where $\epsilon_p = \|\mathbf{X}_p\mathbf{H}_p - \mathbf{B}_p\mathbf{Z}\mathbf{H}_p\|_{\mathbf{F}}^2$. The optimal $\boldsymbol{\gamma}$ can be obtained according to Cauchy–Schwarz inequality as

$$\boldsymbol{\gamma} = \frac{\boldsymbol{\delta}}{\sum_{p=1}^{v}\delta_p}, \tag{14}$$

where $\boldsymbol{\delta} = [\delta_1, \delta_2, \ldots, \delta_v]$ and $\delta_i = \frac{1}{\epsilon_i}$. The time complexity of optimizing $\boldsymbol{\gamma}$ is $\mathcal{O}(nmd)$.

The entire optimization process for solving Eq. (3) can be summarized in Algorithm. 1.

Algorithm 1 FIMVC-VIA

Input: v views incomplete dataset $\{\mathbf{X}_p\}_{p=1}^v$, the missing index $\{\mathbf{H}_p\}_{p=1}^v$ and the number of cluster k.

1: Initialize $\{\mathbf{B}_p\}_{p=1}^v$, $\mathbf{Z}$ and $\gamma_p = \frac{1}{v}$.
2: **repeat**
3: Update $\{\mathbf{B}_p\}_{p=1}^v$ by solving Eq. (4);
4: Update $\mathbf{Z}$ by solving Eq. (6);
5: Update $\boldsymbol{\gamma}$ solving Eq. (13);
6: **until** converged.

Output: Performing SVD on $\mathbf{Z}$.

C. Complexity Analysis

In the optimization process, optimizing $\mathbf{B}_p$ at each iteration costs $\mathcal{O}(nmd_p + m^2d_p)$. When learning the consensus anchor graph $\mathbf{Z}$, the time complexity is $\mathcal{O}(nmd)$. The time cost of updating view weight $\boldsymbol{\gamma}$ is $\mathcal{O}(nmd)$. Therefore, the time complexity in the optimization phase is $\mathcal{O}(nmd + m^2d)$ per iteration. Notice that here m and d are much smaller than n. Therefore, the proposed optimization algorithm is linear complexity with respect to samples. After optimization, performing SVD on $\mathbf{Z}$ and performing k-means needs $\mathcal{O}(nm^2)$, which is also a linearly growing complexity with n. Briefly, our proposed FIMVC-VIA handles IMC problems with linear computational complexity.

The main space consumption of FIMVC-VIA comes from matrices $\mathbf{B}_p \in \mathbb{R}^{d_p\times m}$, $\mathbf{A}_p \in \mathbb{R}^{d_p\times n}$ and $\mathbf{Z} \in \mathbb{R}^{m\times n}$. Accordingly, the space complexity of our FIMVC-VIA is $\mathcal{O}(nm + nd + md)$, which is linear with data size.

D. Convergence Analysis

In this section, we perform a theoretical analysis of the convergence of our proposed algorithm. The objective function of FIMVC-VIA can be defined as follows,

$$\mathcal{J}\left(\boldsymbol{\gamma}, \{\mathbf{B}_p\}_{p=1}^v, \mathbf{Z}\right) = \sum_{p=1}^{v}\gamma_p^2\|\mathbf{X}_p\mathbf{H}_p - \mathbf{B}_p\mathbf{Z}\mathbf{H}_p\|_{\mathbf{F}}^2 + \mu\|\mathbf{Z}\|_{\mathbf{F}}^2, \tag{15}$$

When considering all variables jointly, the function Eq. (15) is non-convex. To solve this problem, we propose to optimize each variable while fixing other variables with a three-step algorithm. First, we define the solution at the t-iteration as $\boldsymbol{\gamma}^{(t)}, \left\{\mathbf{B}_p^{(t)}\right\}_{p=1}^v, \mathbf{Z}^{(t)}$.

i) Given $\boldsymbol{\gamma}^{(t)}, \mathbf{Z}^{(t)}$, the optimal $\{\mathbf{B}_p\}_{p=1}^v$ can be analytically obtained. Suppose the obtained optimal solution be $\left\{\mathbf{B}_p^{(t+1)}\right\}_{p=1}^v$. We have

$$\mathcal{J}\left(\boldsymbol{\gamma}^{(t)}, \left\{\mathbf{B}_p^{(t)}\right\}_{p=1}^v, \mathbf{Z}^{(t)}\right) \geq \mathcal{J}\left(\boldsymbol{\gamma}^{(t)}, \left\{\mathbf{B}_p^{(t+1)}\right\}_{p=1}^v, \mathbf{Z}^{(t)}\right). \tag{16}$$

ii) Given $\boldsymbol{\gamma}^{(t)}, \left\{\mathbf{B}_p^{(t+1)}\right\}_{p=1}^v$, the optimal $\mathbf{Z}$ can be analytically obtained. Suppose the obtained optimal solution be $\mathbf{Z}^{(t+1)}$. We have

$$\mathcal{J}\left(\boldsymbol{\gamma}^{(t)}, \left\{\mathbf{B}_p^{(t+1)}\right\}_{p=1}^v, \mathbf{Z}^{(t)}\right) \geq \mathcal{J}\left(\boldsymbol{\gamma}^{(t)}, \left\{\mathbf{B}_p^{(t+1)}\right\}_{p=1}^v, \mathbf{Z}^{(t+1)}\right). \tag{17}$$

iii) Given $\left\{\mathbf{B}_p^{(t+1)}\right\}_{p=1}^v, \mathbf{Z}^{(t+1)}$, the optimal $\boldsymbol{\gamma}$ can be analytically obtained. Suppose the obtained optimal solution be $\boldsymbol{\gamma}^{(t+1)}$. We have

$$\mathcal{J}\left(\boldsymbol{\gamma}^{(t)}, \left\{\mathbf{B}_p^{(t+1)}\right\}_{p=1}^v, \mathbf{Z}^{(t+1)}\right) \geq \mathcal{J}\left(\boldsymbol{\gamma}^{(t+1)}, \left\{\mathbf{B}_p^{(t+1)}\right\}_{p=1}^v, \mathbf{Z}^{(t+1)}\right). \tag{18}$$

Together with Eq. (16), (17) and (18), we have that

$$\mathcal{J}\left(\boldsymbol{\gamma}^{(t)}, \left\{\mathbf{B}_p^{(t)}\right\}_{p=1}^v, \mathbf{Z}^{(t)}\right) \geq \mathcal{J}\left(\boldsymbol{\gamma}^{(t+1)}, \left\{\mathbf{B}_p^{(t+1)}\right\}_{p=1}^v, \mathbf{Z}^{(t+1)}\right). \tag{19}$$

which demonstrates that the objective function of FIMVC-VIA decreases monotonically with increasing number of iterations. Moreover, the objective function in Eq. (15) is lower bounded by zero. Therefore, our proposed FIMVC-VIA can be proven to converge to a local minimum.

IV. Experiments

In this section, experiments conducted on seven widely used datasets demonstrates the effectiveness of our FIMVC-VIA. To further verify the reliability of our method, we design experiments to analyse the parameter sensitivity and convergence of the algorithm.

A. Experimental Settings

1) Datasets description: We evaluate the experiments on seven widely-used datasets including ORL[1], NGs[2], WebKB[3], STL10[4], Cifar10, Cifar100[5] and MNIST[6]. The datasets are summarized in Table I.

ORL contains frontal images of 40 objects in 10 different scenarios. For the same object, different lighting settings and facial details result in differences between samples.

NGs is a subset of the NG20 dataset with three different preprocessing, including supervised mutual information preprocessing, partitioning around medoids preprocessing and unsupervised mutual information preprocessing [68].

WebKB is a web page dataset described by two views of content and citations, which were collected from four universities websites [69].

STL10 comprises 10 types of animal and transport images, each of which is 96×96 pixels. We extract their features on three views through ResNet18, ResNet50, and DenseNet121.

[1]http://www.cad.zju.edu.cn/home/dengcai/Data/FaceData.html
[2]https://lig-membres.imag.fr/grimal/data.html
[3]http://www.cs.umd.edu/sen/lbc-proj/LBC.html
[4]https://cs.stanford.edu/acoates/stl10/
[5]http://www.cs.toronto.edu/kriz/cifar.html
[6]http://yann.lecun.com/exdb/mnist/

TABLE I: Detail of the employed multi-view datasets.

Dataset	ORL	NGs	WebKB	STL10	Cifar10	Cifar100	MNIST
#Size	400	500	1051	13000	50000	50000	60000
#Classes	40	5	2	10	10	100	10
#Views	3	3	2	3	3	3	3
#View1	Intensity(4096)	SMI(500)	Content(1840)	DenseNet(2048)	DenseNet(2048)	DenseNet(2048)	DenseNet(342)
#View2	LBP(3304)	PAM(500)	Citations(3000)	ResNet18(512)	ResNet18(512)	ResNet18(512)	ResNet101(64)
#View3	Gabor(6750)	UMI(500)	-	ResNet50(1024)	ResNet50(1024)	ResNet50(1024)	ResNet50(1024)

TABLE II: Clustering performance on seven incomplete multi-view datasets. 'N/A' means out of the CPU memory. It is shown in bold with brackets for the best results and in bold for the second best results.

Datasets	BSV	MIC	MKKM-IK	DAIMC	APMC	PIC	UEAF	EEIMVC	FLSD	V^3H	IMVC-CBG	Proposed
ACC												
ORL	24.32±0.89	37.56±1.66	59.80±2.44	68.03±2.32	65.58±1.91	69.08±1.71	60.25±2.50	**73.24±2.54**	48.09±1.85	67.03±1.45	69.32±2.37	**[76.36±2.79]**
NGs	41.13±2.03	20.92±0.46	79.63±0.09	**89.41±0.01**	**89.41±0.01**	82.20±0.22	89.49±0.03	77.84±0.12	84.19±0.03	79.77±0.45	88.95±0.02	**[89.76±0.01]**
WebKB	57.08±2.39	63.88±0.76	68.04±0.00	N/A	**85.36±0.03**	71.53±0.00	82.64±0.67	61.67±3.49	78.21±0.00	74.69±10.53	84.52±0.02	**[91.51±0.00]**
STL10	11.22±0.09	N/A	**75.83±0.26**	22.96±1.84	26.98±1.30	28.73±0.16	73.57±2.71	46.71±2.34	64.71±4.72	18.41±0.83	55.59±0.03	**[75.95±0.00]**
Cifar10	N/A	N/A	N/A	90.81±0.45	N/A	N/A	N/A	N/A	N/A	N/A	**96.19±0.13**	**[96.56±0.00]**
Cifar100	N/A	N/A	N/A	89.71±1.00	N/A	N/A	N/A	N/A	N/A	N/A	**93.09±1.18**	**[98.86±0.67]**
MNIST	N/A	N/A	N/A	97.57±0.31	N/A	N/A	N/A	N/A	N/A	N/A	**98.10±0.06**	**[98.70±0.00]**
NMI												
ORL	48.49±0.90	56.44±1.00	75.95±1.33	82.89±1.06	80.20±0.82	83.21±0.70	76.16±1.25	**85.37±1.32**	67.91±1.28	81.05±0.61	81.17±1.30	**[88.08±1.31]**
NGs	20.24±1.37	2.37±0.49	63.16±0.11	73.36±0.03	73.36±0.03	65.59±0.16	**73.60±0.08**	57.23±0.18	64.22±0.04	59.07±0.39	73.05±0.06	**[75.41±0.05]**
WebKB	1.80±0.86	3.29±0.29	4.06±0.00	N/A	**47.88±0.10**	1.67±0.00	23.52±0.97	3.57±0.56	4.56±4.14	21.68±11.13	37.27±0.10	**[48.89±0.00]**
STL10	0.15±0.02	N/A	**[60.22±0.31]**	5.05±1.22	11.04±0.95	14.22±0.10	55.71±0.93	29.21±3.53	47.34±2.83	5.92±0.74	27.15±0.14	**57.28±0.00**
Cifar10	N/A	N/A	N/A	90.47±0.55	N/A	N/A	N/A	N/A	N/A	N/A	**90.89±0.27**	**[91.38±0.00]**
Cifar100	N/A	N/A	N/A	98.26±0.16	N/A	N/A	N/A	N/A	N/A	N/A	**98.63±0.29**	**[99.78±0.11]**
MNIST	N/A	N/A	N/A	93.89±0.53	N/A	N/A	N/A	N/A	N/A	N/A	**94.94±0.13**	**[96.17±0.00]**
Purity												
ORL	26.80±0.92	40.81±1.40	62.79±2.11	71.82±1.79	69.24±1.48	72.24±1.36	63.90±1.90	**76.09±2.19**	50.88±1.72	70.22±1.09	69.32±1.86	**[79.15±2.29]**
NGs	43.15±1.51	21.30±0.42	79.63±0.09	89.41±0.01	89.41±0.01	82.36±0.22	**89.49±0.03**	77.84±0.12	84.19±0.03	79.77±0.45	88.95±0.02	**[89.76±0.01]**
WebKB	78.12±0.00	78.12±0.00	78.12±0.00	N/A	**90.17±0.02**	78.12±0.00	82.91±0.09	78.12±0.00	78.72±1.15	81.38±3.25	84.52±0.02	**[91.51±0.00]**
STL10	11.33±0.09	N/A	**75.83±0.26**	23.22±1.72	27.50±1.26	29.38±0.13	73.60±2.66	46.94±2.22	64.81±4.63	18.62±0.80	55.59±0.03	**[75.95±0.00]**
Cifar10	N/A	N/A	N/A	95.81±0.45	N/A	N/A	N/A	N/A	N/A	N/A	**96.19±0.26**	**[96.56±0.00]**
Cifar100	N/A	N/A	N/A	92.60±0.54	N/A	N/A	N/A	N/A	N/A	N/A	**94.98±0.82**	**[99.02±0.53]**
MNIST	N/A	N/A	N/A	97.57±0.31	N/A	N/A	N/A	N/A	N/A	N/A	**98.10±0.16**	**[98.70±0.00]**
Fscore												
ORL	9.01±0.69	17.30±1.18	46.32±2.50	56.84±2.87	50.70±2.55	57.78±1.78	42.53±2.74	**63.67±2.85**	31.17±2.00	54.27±1.40	46.26±3.67	**[68.25±3.25]**
NGs	32.39±1.08	32.90±0.19	68.72±0.07	80.28±0.02	80.28±0.02	72.51±0.18	**80.55±0.06**	63.75±0.14	71.80±0.05	65.45±0.50	79.49±0.03	**[80.72±0.02]**
WebKB	60.55±1.28	61.10±0.61	64.64±0.00	N/A	**84.05±0.03**	73.56±0.00	78.51±0.73	62.75±2.14	79.40±0.00	71.82±0.00	83.01±0.01	**[88.66±0.00]**
STL10	11.76±0.02	N/A	57.47±0.18	13.20±0.75	18.10±0.58	21.62±0.14	**57.63±1.38**	29.21±1.21	50.68±3.05	17.05±0.10	34.53±0.16	**[59.44±0.00]**
Cifar10	N/A	N/A	N/A	92.16±0.68	N/A	N/A	N/A	N/A	N/A	N/A	**92.67±0.13**	**[93.35±0.00]**
Cifar100	N/A	N/A	N/A	90.82±0.94	N/A	N/A	N/A	N/A	N/A	N/A	**90.87±2.97**	**[98.88±0.63]**
MNIST	N/A	N/A	N/A	95.28±0.57	N/A	N/A	N/A	N/A	N/A	N/A	**96.29±0.28**	**[97.44±0.00]**

TABLE III: Time comparison of different IMC methods on seven partial datasets. 'N/A' means out of CPU memory.

Methods	ORL		NGS		WebKB		STL10		Cifar10		Cifar100		MNIST	
	Time(s)	Speed-up	Time(s)	Speed-up	Time(s)	Speed-up	Time(s)	Speed-up	Time(s)	Speed-up	Time(s)	Speed-up	Time(s)	Speed-up
BSV	0.10	1.00×	0.06	1.00×	0.13	1.00×	66.97	11.87×	N/A	N/A	N/A	N/A	N/A	N/A
MIC	418.26	4182.60×	143.83	2,397.16×	335.62	2581.69×	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
MKKM-IK	0.55	5.50×	0.50	8.30×	3.12	24.00×	1665.99	295.39×	N/A	N/A	N/A	N/A	N/A	N/A
DAIMC	1205.49	12054.90×	0.18	3.00×	N/A	N/A	574.98	101.95×	28193.89	1237.66×	24794.67	537.73×	5588.82	284.56×
APMC	0.47	4.70×	0.18	3.00×	0.21	1.62×	63.71	11.30×	N/A	N/A	N/A	N/A	N/A	N/A
PIC	0.28	2.80×	0.28	4.67×	1.27	9.77×	3343.90	592.89×	N/A	N/A	N/A	N/A	N/A	N/A
UEAF	14.19	141.90×	2.25	37.50×	5.72	44.00×	4104.64	727.77×	N/A	N/A	N/A	N/A	N/A	N/A
EEIMVC	0.47	4.70×	0.14	2.33×	0.19	1.46×	66.99	11.88×	N/A	N/A	N/A	N/A	N/A	N/A
FLSD	15.62	156.20×	3.30	55.00×	7.23	55.62×	1973.36	349.89×	N/A	N/A	N/A	N/A	N/A	N/A
V^3H	87.33	873.30×	14.32	238.67×	31.64	243.38×	44590.51	7906.12×	N/A	N/A	N/A	N/A	N/A	N/A
IMVC-CBG	2.95	29.50×	1.31	21.83×	0.63	4.85×	17.10	3.03×	749.99	32.92×	800.05	17.35×	549.08	27.96×
Proposed	1.52	15.20×	0.23	3.83×	0.23	1.77×	5.64	1.00×	22.78	1.00×	46.11	1.00×	19.64	1.00×

Cifar10 consists of 50,000 tiny images that can be divided into 10 mutually exclusive classes. Similar to STL10, we extract its features on three different views.

Cifar100 is a dataset similar to Cifar10, where images can be tagged with 100 labels.

MNIST is a handwritten dataset containing the numbers 0 to 9, which is a subset of a larger dataset provided by NIST.

For the datasets mentioned above, we remove some instances on each view randomly to get their incomplete versions. Specifically, with the principle that each instance is present in at least one view, we generate missing datasets at missing rates in intervals of 0.1 from 0.1 to 0.9.

2) Compared methods: The following eleven SOTA IMC methods are compared with our proposed method in the experiment:

BSV [70]: It performs spectral clustering each view separately with mean value filling in the incomplete part, then reports the best single view result.

MIC [30]: This work first learns the latent representation from each view and optimize a common clustering representation then.

MKKM-IK [51]: It is an incomplete multi-kernel method which imputes missing part and conducts kernel k-means algorithm together.

DAIMC [58]: This algorithm introduces a view-specific weight matrix to address the missing view problem and aligns the basis matrices.

APMC [62]: This work proposes to use samples present in all views as anchors and perform spectral clustering to get the final result.

PIC [60]: In this paper, the missing view problem is solved by padding the similarity matrix.

UEAF [32]: It recovers the missing views and learns the unified clustering representation simultaneously.

EEIMVC [48]: This algorithm first generates the low dimensional base feature matrices, then imputes them and optimizes a consensus feature matrix in a unified framework.

FLSD [59]: It learns view-specific latent representations and seeks a shared clustering representation based on a semantic consistency constraint jointly.

V^{3}H [61]: Inspired by genetics, it explores the consistent part and unique par among incomplete views.

IMVC-CBG [24]: It proposes to handle IMC problem with a scalable anchor graph framework for the first time.

For the compared IMC methods, we set their parameters as their recommended range. In FIMVC-VIA, we search μ in $[0.001, 0.1, 1, 10, 100, 1000]$ and m in $[k, 2k, 3k, 5k, 7k]$ where m is the anchor number. Four popular metrics, including accuracy (ACC), normalized mutual information (NMI), Purity and Fscore, are employed to evaluate the experimental results. Furthermore, we run each algorithm 20 times and report the average performance and standard deviation. All the experiments are conducted on a desktop with Intel Core i9-10900X CPU and 64G RAM, MATLAB 2019b(64-bit).

B. Experimental Results

Table II compares the ACC, NMI, Purity, and Fscore of the aforementioned baseline methods. In order to further illustrate the capability of FIMVC-VIA for processing partial data, we plot the ACC, NMI, Purity and Fscore of distinct methods on incomplete datasets varying in missing rates in Figure 2, Figure 3, Figure 4, Figure 5. From the table and these figures, we have the following conclusions:

- In comparison to other SOTA IMC approaches, the proposed method consistently provides the best performance. For example, our FIMVC-VIA further obtains a better performance of 4.3%, 0.4%, 7.2%, 0.2%, 0.4%, 6.2%, 0.6% in terms of ACC than the second-best method on seven datasets. Additionally, the optimal performance on other metrics can also show the superior effectiveness of our algorithm.
- Compared to the previous graph or kernel-based IMC methods (MKKM-IK, APMC, PIC, EEIMVC, FLSD), our proposed FIMVC-VIA shows optimal performance on all metrics. Moreover, our anchor-based algorithm can handle large-scale datasets well. From the above experimental results, our algorithm is effective in applying anchor graphs to solve IMC problems.
- Compared to IMVC-CBG, which learns inter-view uniform anchors, our approach has better results on all metrics for all datasets. This result demonstrates the effectiveness of learning independent anchor points on each view to explore the view-specific information.
- As seen from the figures, the performance of almost all algorithms drops as the missing rate grows. In contrast to other IMC algorithms, FIMVC-VIA attains the best performance on all datasets in every metrics despite high missing rates. Consequently, FIMVC-VIA addresses partial data efficiently.

In conclusion, FIMVC-VIA exhibits clustering performance that far exceeds expectations on all datasets and works well when dealing with large-scale datasets. Therefore, the above results exhibit the effectiveness of our proposed FIMVC-VIA.

C. Running Time Analysis

Table III displays the average execution times of the aforementioned algorithms across all benchmark datasets with various missing rates. We have the following conclusion with the observation of Table III:

- Almost all methods consume more runtime as the size of the datasets increases. When the datasets are close in size, those with larger data dimensions tend to take more time. For example, MNIST is larger than Cifar10 in size, but consumes less time instead because the dimensions are smaller.
- APMC is an anchor-based method which takes the samples existed in all view as anchors. When dealing with large-scale datasets (e.g., STL10), APMC is far less fast than our method. Anchor selection with much smaller data size and clever transformation of the missing index matrix make our method more efficient than APMC.
- Our FIMVC-VIA takes much less run-time on ORL, NGs, WebKB, and STL10 than the other IMC algorithms, indicating its efficiency in computation. All other comparison methods suffer from memory shortage regarding Cifar10, Cifar100 and MNIST. DAIMC and IMVC-CBG, the only two available approach, require far more time compared to our proposed method. This demonstrates the efficiency of our FIMVC-VIA.

D. Sensitivity Analysis

Two hyper-parameters are employed in our method, μ is the coefficient on the sparsity regularization term and m is the number of anchor points. We conduct a comparative experiment on the ORL, NGs, STL10 and Cifar100 datasets to illustrate the impact of these two parameters on performance. For ORL, our method works better when μ is greater than 1 and less than 100, and is not affected much by the variation of m. FIMVC-VIA works well on NGs dataset when μ takes values between 0.001 and 1, and m has little effect on it. As shown in Figure 7(c) and 7(d), we have optimal results of our method on the STL10 and Cifar100 with μ taking the value of 0.001. Therefore, variations in m have little effect on ACC when fixing μ and the ACC under the same m will be affected by the μ.

E. Qualified Study

In addition, the representation of the learned data for different iterations on the NGs dataset is plotted in Figure 8. At the first iteration, the relationship between all sample points is very unclear, and the points are scattered in the space without any pattern. As iterative optimization proceeds, the cluster structure of the data becomes clearer and more easily

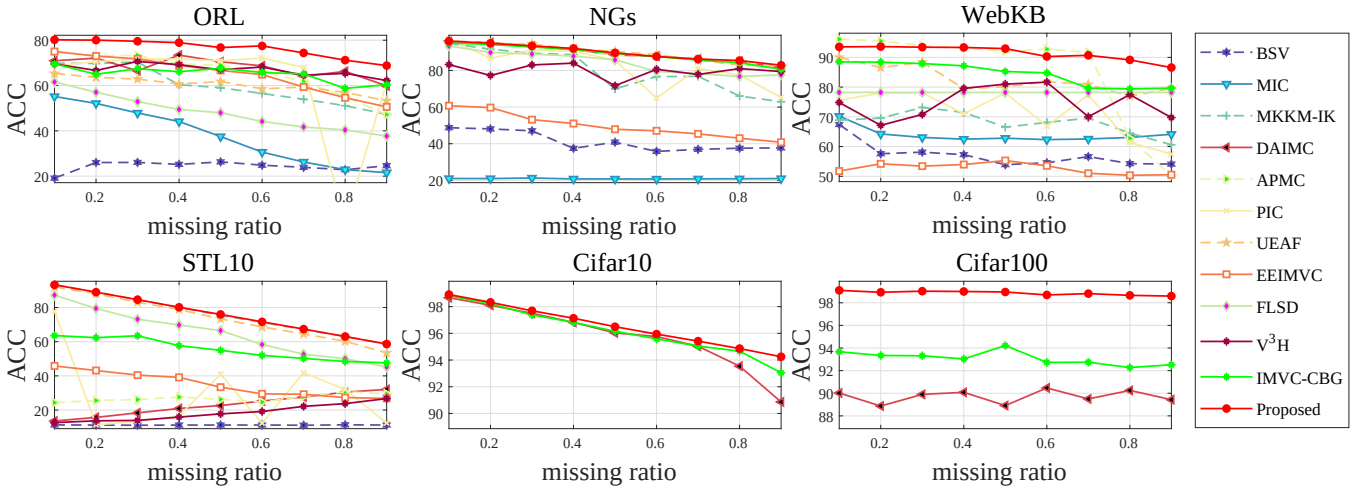


Fig. 2: ACC for clustering results on benchmark datasets with different missing rates.

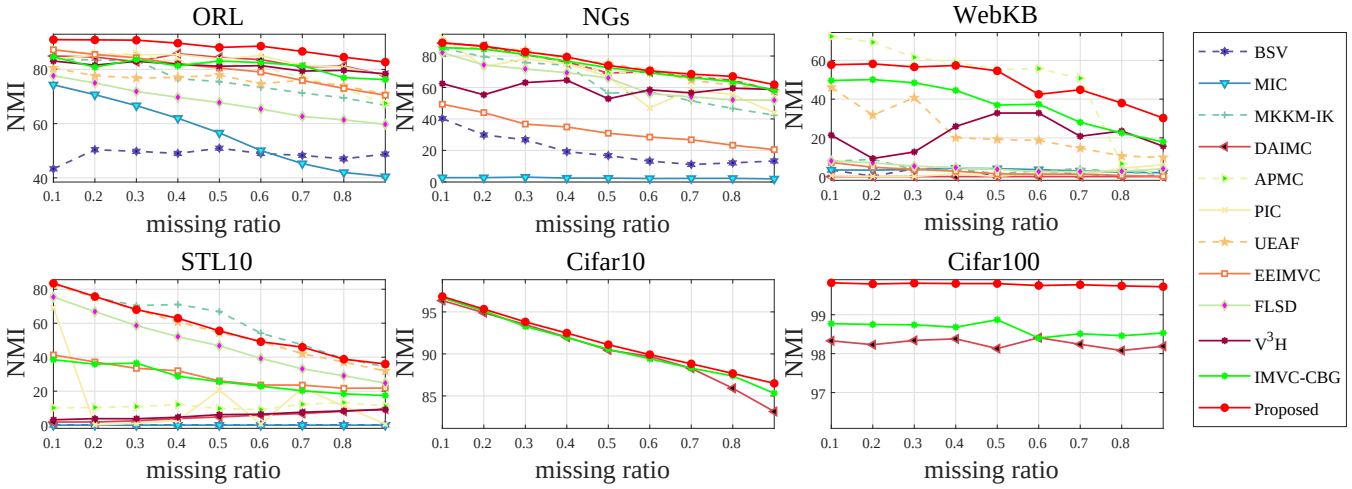


Fig. 3: NMI for clustering results on benchmark datasets with different missing rates.

TABLE IV: Comparison of separate anchor graphs generation strategy and consensus anchor graph generation strategy.

Metrics	Strategy	Datasets						
		ORL	NGs	WebKB	STL10	Cifar10	Cifar100	MNIST
ACC	Separate	68.61±2.37	66.66±0.36	77.75±0.03	56.81±0.03	85.28±0.12	92.83±1.19	93.78±0.00
	Consensus	**76.36±2.79**	**89.76±0.01**	**91.51±0.00**	**75.95±0.00**	**96.56±0.00**	**98.86±0.67**	**98.70±0.00**
NMI	Separate	82.62±1.26	38.38±0.43	14.52±0.03	39.59±0.07	77.67±0.11	97.75±0.51	90.03±0.00
	Consensus	**88.08±1.31**	**75.41±0.05**	**48.89±0.00**	**57.28±0.00**	**91.38±0.00**	**99.78±0.11**	**96.17±0.00**
Purity	Separate	71.63±1.98	67.02±0.36	80.23±0.02	56.86±0.03	85.55±0.12	93.86±1.07	93.78±0.00
	Consensus	**79.15±2.29**	**89.76±0.01**	**91.51±0.00**	**75.95±0.00**	**96.56±0.00**	**99.02±0.53**	**98.70±0.00**
Fscore	Separate	57.46±2.68	49.41±0.32	73.76±0.04	39.79±0.06	77.01±0.17	92.50±1.34	90.21±0.00
	Consensus	**68.25±3.25**	**80.72±0.02**	**88.66±0.00**	**59.44±0.00**	**93.35±0.00**	**98.88±0.63**	**97.44±0.00**

distinguishable. At the 10-th iteration, it can be seen from the Figure 8(d) that the sample points in the same cluster are compactly distributed, while the distance between sample points in different clusters is large.

F. Ablation Study

In this section, we conduct comparative experiments to demonstrate the superiority of constructing a consistent anchor graph. In the designed experiments, we learn independent anchor graphs on each view, and finally perform the clustering process on the anchor graph composed of their averages. The clustering performance comparison of separate anchor graphs generation strategy and our consensus anchor graph generation strategy are shown in Table IV. As can be seen, our proposed method significantly outperforms the separate competitors, which demonstrates the superiority of our consensus anchor graph generation strategy.

G. Convergence Analysis

According to [71] and Section III-D, FIMVC-VIA is guaranteed to converge to a local minimum in theory. We further

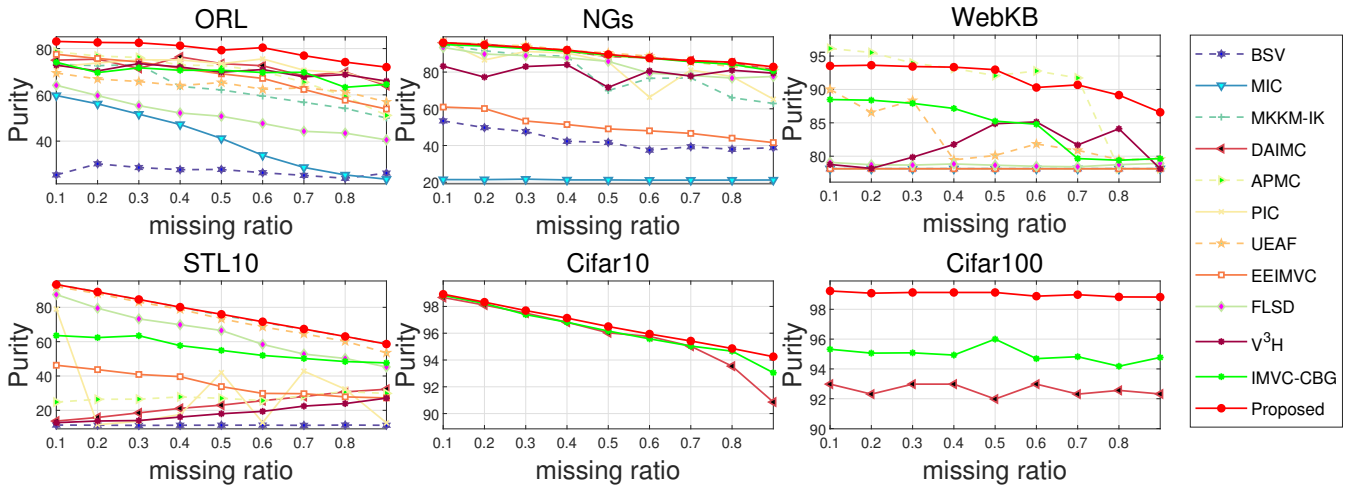


Fig. 4: Purity for clustering results on benchmark datasets with different missing rates.

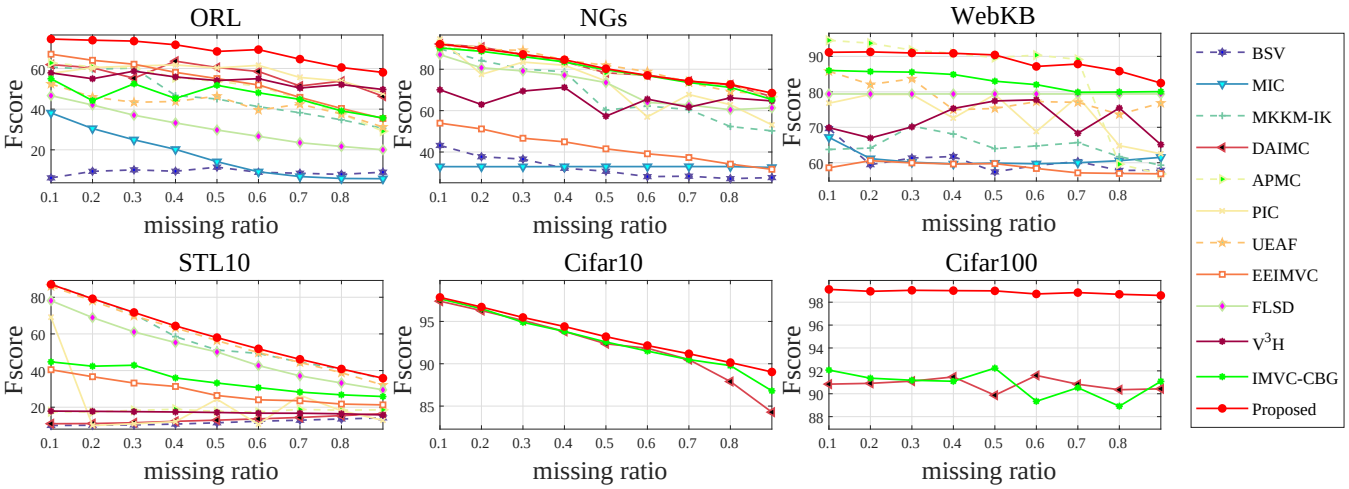


Fig. 5: Fscore for clustering results on benchmark datasets with different missing rates.

give the evolution of the objective function value with iterations in Figure 6 to indicate the convergence of our proposed FIMVC-VIA. It is shown in the figure that the objective function value decreases monotonically with iteration, verifying the convergence of our algorithm.

V. Conclusion

In this paper, we propose a novel method termed as Fast Incomplete Multi-view Clustering with View-independent Anchors (FIMVC-VIA). Different from existing graph-based incomplete multi-view clustering methods, we construct an anchor graph instead of a full graph to measure the distance between existing samples and each anchor. Meanwhile, we learn view-independent anchors to explore the view-specific information and construct a consensus anchor graph for Consistency. Extensive experiments performed on seven benchmark datasets demonstrate that our method could handle the IMC problem effectively and efficiently in comparison with the state-of-the-art approaches. In the future work, we will consider the alignment of anchor points between views and explore the way to determine the optimal number of anchor points.

Acknowledgment

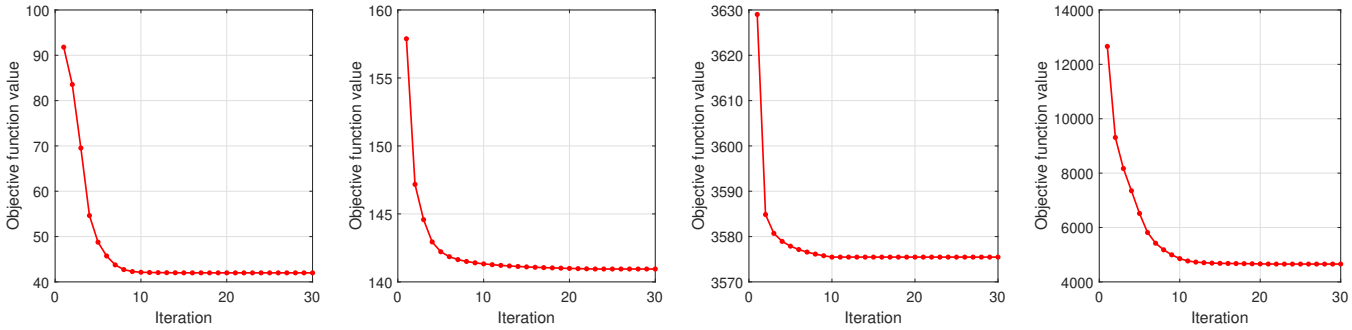
This work was supported by the National Key R&D Program of China (project no. 2020AAA0107100), the National Natural Science Foundation of China (project no. 61922088, 61906020, 61872371 and 62006237) and the Outstanding Talents of "Ten Thousand Talents Plan" in Zhejiang Province (project no. 2018R51001).

References

[1] X. Cai, F. Nie, and H. Huang, "Multi-view k-means clustering on big data," in *Twenty-Third International Joint conference on artificial intelligence*, 2013.

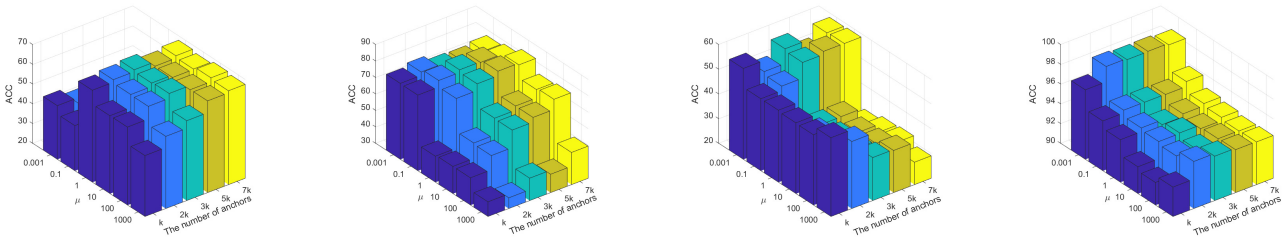
[2] H. Gao, F. Nie, X. Li, and H. Huang, "Multi-view subspace clustering," in *Proceedings of the IEEE international conference on computer vision*, 2015.

[3] Z. Huang, P. Hu, J. T. Zhou, J. Lv, and X. Peng, "Partially view-aligned clustering," *Advances in Neural Information Processing Systems*, vol. 33, pp. 2892–2902, 2020.



(a) ORL (b) NGs (c) STL10 (d) Cifar100

Fig. 6: The objectives of our proposed method on four benchmark datasets.



(a) ORL (b) NGs (c) STL10 (d) Cifar100

Fig. 7: Sensitivity analysis of μ and m and convergence analysis of our method on four datasets.

(a) 1-st iteration (b) 3-rd iteration (c) 5-th iteration (d) 10-th iteration

Fig. 8: Illustrations of the learned data distribution with t-sne algorithm on NGs datasets.

[4] X. Wan, J. Liu, W. Liang, X. Liu, Y. Wen, and E. Zhu, "Continual multi-view clustering," in *Proceedings of the 30th ACM International Conference on Multimedia*, ser. MM '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 3676–3684. [Online]. Available: https://doi.org/10.1145/3503161.3547864

[5] R. Li, C. Zhang, Q. Hu, P. Zhu, and Z. Wang, "Flexible multi-view representation learning for subspace clustering." in *Proc. of IJCAI*, 2019.

[6] X. Peng, Z. Huang, J. Lv, H. Zhu, and J. T. Zhou, "Comic: Multi-view clustering without parameter selection," in *Proc. of ICML*, 2019.

[7] Z. Kang, X. Zhao, C. Peng, H. Zhu, J. T. Zhou, X. Peng, W. Chen, and Z. Xu, "Partition level multiview subspace clustering," *Neural Networks*, 2020.

[8] Q. Qiang, B. Zhang, F. Wang, and F. Nie, "Fast multi-view discrete clustering with anchor graphs," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 11, 2021, pp. 9360–9367.

[9] L. Li, S. Wang, X. Liu, E. Zhu, L. Shen, K. Li, and K. Li, "Local sample-weighted multiple kernel clustering with consensus discriminative graph," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–14, 2022.

[10] A. Kumar, P. Rai, and H. Daume, "Co-regularized multi-view spectral clustering," in *Advances in neural information processing systems*, 2011.

[11] C.-K. Lee and T.-L. Liu, "Guided co-training for multi-view spectral clustering," in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2016, pp. 4042–4046.

[12] S. Huang, Z. Xu, I. W. Tsang, and Z. Kang, "Auto-weighted multi-view co-clustering with bipartite graphs," *Information Sciences*, vol. 512, pp. 18–30, 2020.

[13] S. Zhou, X. Liu, M. Li, E. Zhu, L. Liu, C. Zhang, and J. Yin, "Multiple kernel clustering with neighbor-kernel subspace segmentation," *IEEE transactions on neural networks and learning systems*, 2019.

[14] Z. Ren, S. X. Yang, Q. Sun, and T. Wang, "Consensus affinity graph learning for multiple kernel clustering," *IEEE Transactions on Cybernetics*, vol. 51, no. 6, pp. 3273–3284, 2020.

[15] M. Brbić and I. Kopriva, "Multi-view low-rank sparse subspace clustering," *Pattern Recognition*, 2018.

[16] X. Wang, Z. Lei, X. Guo, C. Zhang, H. Shi, and S. Z. Li, "Multi-view subspace clustering with intactness-aware similarity," *Pattern Recognition*, 2019.

[17] J. Zhang, L. Li, S. Wang, J. Liu, Y. Liu, X. Liu, and E. Zhu, "Multiple kernel clustering with dual noise minimization," in *Proceedings of*

the 30th ACM International Conference on Multimedia, ser. MM '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 3440–3450.

[18] H. Zhao, Z. Ding, and Y. Fu, "Multi-view clustering via deep matrix factorization," in *Proc. of AAAI*, 2017.

[19] 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*, 2018.

[20] M. Hu and S. Chen, "One-pass incomplete multi-view clustering," in *Proc. of AAAI*, 2019.

[21] S. Wang, X. Liu, X. Zhu, P. Zhang, Y. Zhang, F. Gao, and E. Zhu, "Fast parameter-free multi-view subspace clustering with consensus anchor guidance," *IEEE Transactions on Image Processing*, 2021.

[22] M. Yang, Y. Li, Z. Huang, Z. Liu, P. Hu, and X. Peng, "Partially view-aligned representation learning with noise-robust contrastive loss," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 1134–1143.

[23] C. Xu, Z. Guan, W. Zhao, H. Wu, Y. Niu, and B. Ling, "Adversarial incomplete multi-view clustering." in *Proc. of IJCAI*, 2019.

[24] S. Wang, X. Liu, L. Liu, W. Tu, X. Zhu, J. Liu, S. Zhou, and E. Zhu, "Highly-efficient incomplete large-scale multi-view clustering with consensus bipartite graph," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2022.

[25] P. Zhang, S. Wang, J. Hu, Z. Cheng, X. Guo, E. Zhu, and Z. Cai, "Adaptive weighted graph fusion incomplete multi-view subspace clustering," *Sensors*, 2020.

[26] M. Li, J. Xia, H. Xu, Q. Liao, X. Zhu, and X. Liu, "Localized incomplete multiple kernel k-means with matrix-induced regularization," *IEEE Transactions on Cybernetics*, 2021.

[27] J. Liu, X. Liu, Y. Zhang, P. Zhang, W. Tu, S. Wang, S. Zhou, W. Liang, S. Wang, and Y. Yang, "Self-representation subspace clustering for incomplete multi-view data," in *Proc. of ACM MM*, 2021.

[28] S.-Y. Li, Y. Jiang, and Z.-H. Zhou, "Partial multi-view clustering," in *Proc. of AAAI*, 2014.

[29] H. Zhao, H. Liu, and Y. Fu, "Incomplete multi-modal visual data grouping." in *Proc. of IJCAI*, 2016.

[30] W. Shao, L. He, and S. Y. Philip, "Multiple incomplete views clustering via weighted nonnegative matrix factorization with $\ell_{2,1}$ regularization," in *Joint European conference on machine learning and knowledge discovery in databases*, 2015.

[31] X. Liu, X. Zhu, M. Li, L. Wang, E. Zhu, T. Liu, M. Kloft, D. Shen, J. Yin, and W. Gao, "Multiple kernel k k -means with incomplete kernels," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[32] J. Wen, Z. Zhang, Y. Xu, B. Zhang, L. Fei, and H. Liu, "Unified embedding alignment with missing views inferring for incomplete multi-view clustering," in *Proc. of AAAI*, 2019.

[33] J. Wen, Z. Zhang, Y. Xu, B. Zhang, L. Fei, and G.-S. Xie, "Cdimc-net: Cognitive deep incomplete multi-view clustering network." in *Proc. of IJCAI*, 2020.

[34] J. Wen, Z. Wu, Z. Zhang, L. Fei, B. Zhang, and Y. Xu, "Structural deep incomplete multi-view clustering network," in *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 2021, pp. 3538–3542.

[35] J. Wen, Z. Zhang, Z. Zhang, Z. Wu, L. Fei, Y. Xu, and B. Zhang, "Dimc-net: Deep incomplete multi-view clustering network," in *Proc. of ACM MM*, 2020.

[36] Y. Lin, Y. Gou, Z. Liu, B. Li, J. Lv, and X. Peng, "Completer: Incomplete multi-view clustering via contrastive prediction," in *Proc. of CVPR*, 2021.

[37] M. Sun, P. Zhang, S. Wang, S. Zhou, W. Tu, X. Liu, E. Zhu, and C. Wang, "Scalable multi-view subspace clustering with unified anchors," in *Proc. of ACM MM*, 2021.

[38] S. Liu, S. Wang, P. Zhang, K. Xu, X. Liu, C. Zhang, and F. Gao, "Efficient one-pass multi-view subspace clustering with consensus anchors," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 7, 2022, pp. 7576–7584.

[39] S. Wang, X. Liu, S. Liu, J. Jin, W. Tu, X. Zhu, and E. Zhu, "Align then fusion: Generalized large-scale multi-view clustering with anchor matching correspondences," *arXiv preprint arXiv:2205.15075*, 2022.

[40] Z. Kang, W. Zhou, Z. Zhao, J. Shao, M. Han, and Z. Xu, "Large-scale multi-view subspace clustering in linear time," in *Proc. of AAAI*, 2020.

[41] M. Li, S. Wang, X. Liu, and S. Liu, "Parameter-free and scalable incomplete multiview clustering with prototype graph," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[42] P. Xu, Z. Deng, K.-S. Choi, L. Cao, and S. Wang, "Multi-view information-theoretic co-clustering for co-occurrence data," in *Proc. of AAAI*, 2019.

[43] S. Du, Z. Liu, Z. Chen, W. Yang, and S. Wang, "Differentiable bi-sparse multi-view co-clustering," *IEEE Transactions on Signal Processing*, 2021.

[44] G. Bisson and C. Grimal, "Co-clustering of multi-view datasets: a parallelizable approach," in *Proc. of ICDM*, 2012.

[45] J. Sun, J. Lu, T. Xu, and J. Bi, "Multi-view sparse co-clustering via proximal alternating linearized minimization," in *Proc. of ICML*, 2015.

[46] X. Liu, S. Zhou, L. Liu, C. Tang, S. Wang, J. Liu, and Y. Zhang, "Localized simple multiple kernel k-means," in *Proc. of ICCV*, 2021.

[47] S. Wang, X. Liu, L. Liu, S. Zhou, and E. Zhu, "Late fusion multiple kernel clustering with proxy graph refinement," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[48] X. Liu, M. Li, C. Tang, J. Xia, J. Xiong, L. Liu, M. Kloft, and E. Zhu, "Efficient and effective regularized incomplete multi-view clustering." *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[49] X. Liu, "Simplemkkm: Simple multiple kernel k-means," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[50] H.-C. Huang, Y.-Y. Chuang, and C.-S. Chen, "Multiple kernel fuzzy clustering," *IEEE Transactions on Fuzzy Systems*, 2012.

[51] X. Liu, Y. Dou, J. Yin, L. Wang, and E. Zhu, "Multiple kernel k -means clustering with matrix-induced regularization," in *Proc. of AAAI*, 2016.

[52] Z. Ma, Z. Kang, G. Luo, L. Tian, and W. Chen, "Towards clustering-friendly representations: Subspace clustering via graph filtering," in *Proc. of ACM MM*, 2020.

[53] C. Zhang, Q. Hu, H. Fu, P. Zhu, and X. Cao, "Latent multi-view subspace clustering," in *Proc. of CVPR*, 2017.

[54] S. Luo, C. Zhang, W. Zhang, and X. Cao, "Consistent and specific multi-view subspace clustering," in *Proc. of AAAI*, 2018.

[55] 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 *Proc. of IJCAI*, 2016.

[56] K. Zhan, C. Zhang, J. Guan, and J. Wang, "Graph learning for multiview clustering," *IEEE transactions on cybernetics*, 2017.

[57] 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*, 2017.

[58] M. Hu and S. Chen, "Doubly aligned incomplete multi-view clustering," in *Proc. of IJCAI*, 2018.

[59] J. Wen, Z. Zhang, Z. Zhang, L. Fei, and M. Wang, "Generalized incomplete multiview clustering with flexible locality structure diffusion," *IEEE transactions on cybernetics*, 2020.

[60] H. Wang, L. Zong, B. Liu, Y. Yang, and W. Zhou, "Spectral perturbation meets incomplete multi-view data," *arXiv preprint arXiv:1906.00098*, 2019.

[61] X. Fang, Y. Hu, P. Zhou, and D. O. Wu, "V ^{3} h: View variation and view heredity for incomplete multiview clustering," *IEEE Transactions on Artificial Intelligence*, 2020.

[62] J. Guo and J. Ye, "Anchors bring ease: An embarrassingly simple approach to partial multi-view clustering," in *Proc. of AAAI*, 2019.

[63] S. Zhou, D. Nie, E. Adeli, J. Yin, J. Lian, and D. Shen, "High-resolution encoder–decoder networks for low-contrast medical image segmentation," *IEEE Transactions on Image Processing*, vol. 29, pp. 461–475, 2019.

[64] Y. Liu, W. Tu, S. Zhou, X. Liu, L. Song, X. Yang, and E. Zhu, "Deep graph clustering via dual correlation reduction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 7, 2022, pp. 7603–7611.

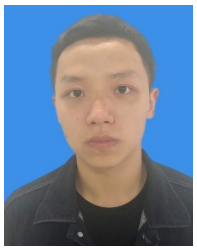
[65] L. Gong, S. Zhou, W. Tu, and X. Liu, "Attributed graph clustering with dual redundancy reduction," in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, L. D. Raedt, Ed. International Joint Conferences on Artificial Intelligence Organization, 7 2022, pp. 3015–3021, main Track. [Online]. Available: https://doi.org/10.24963/ijcai.2022/418

[66] L. Li, Z. Wan, and H. He, "Incomplete multi-view clustering with joint partition and graph learning," *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[67] M. Yang, Y. Li, P. Hu, J. Bai, J. C. Lv, and X. Peng, "Robust multi-view clustering with incomplete information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[68] S. F. Hussain, G. Bisson, and C. Grimal, "An improved co-similarity measure for document clustering," in *2010 Ninth International Conference on Machine Learning and Applications*. IEEE, 2010, pp. 190–197.

[69] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proceedings of the eleventh annual conference on Computational learning theory*, 1998.
[70] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in neural information processing systems*, 2002.
[71] J. C. Bezdek and R. J. Hathaway, "Convergence of alternating optimization," *Neural, Parallel Sci. Comput.*, 2003.



Suyuan Liu is pursuing his M.S. degree in National University of Defense Technology (NUDT), China. His current research interests include multiple-view learning and scalable clustering.

Xinwang Liu received his PhD degree from National University of Defense Technology (NUDT), China, in 2013. He is now Professor at School of Computer, NUDT. His current research interests include kernel learning, multi-view clustering and unsupervised feature learning. Dr. Liu has published 100+ peer-reviewed papers, including those in highly regarded journals and conferences such as IEEE T-PAMI, IEEE T-KDE, IEEE T-IP, IEEE T-NNLS, IEEE T-MM, IEEE T-IFS, ICML, NeurIPS, CVPR, ICCV, AAAI, IJCAI, etc. He is an Associate Editor of IEEE T-NNLS and Information Fusion Journal. More information can be found at https://xinwangliu.github.io/.

Siwei Wang is pursuing his P.H.D degree in National University of Defense Technology (NUDT), China. His current research interests include unsupervised multiple-view learning, scalable clustering and deep unsupervised learning. He has published several papers and served as PC member/ Reviewer in top journals and conferences such as IEEE TKDE/TNNLS/TIP/TCYB/TMM, ICML/CVPR/ECCV/ICCV/AAAI/IJCAI, etc.

Xin Niu received the B.S. degree in computer science and technology from the National University of Defense Technology, Changsha, China, in 2006, and the Ph.D degree in Geoinformatics in 2012 from the Royal Institute of Technology–KTH, Stockholm, Sweden. Since 2020, he has been an Associate Professor in College of Computer Science and Technology in National University of Defense Technology, Changsha, China. His research interests include machine learning, computer architecture and remote sensing.

En Zhu received his PhD degree from National University of Defense Technology (NUDT), China. He is now Professor at School of Computer Science, NUDT, China. His main research interests are pattern recognition, image processing, machine vision and machine learning. Dr. Zhu has published 60+ peer-reviewed papers, including IEEE T-CSVT, IEEE T-NNLS, PR, AAAI, IJCAI, etc. He was awarded China National Excellence Doctoral Dissertation.