Information Sciences 751 (2026) 123588

Contents lists available at ScienceDirect

Information Sciences

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

Adaptive representation learning framework for incomplete multi-view clustering☆

Yawen He, Mingjing Du*

Jiangsu Key Laboratory of Educational Intelligent Technology, School of Artificial Intelligence and Computer Science, Jiangsu Normal University, Xuzhou, 221116, Jiangsu, China

HIGHLIGHTS

- We introduce ARLIMVC to unify recovery, alignment, fusion, and clustering.
- The mutual information-guided recovery module transfers weighted graphs to missing views.
- The dual-constraint module aligns views and compacts local neighborhoods.
- The KAN fusion module learns instance-specific weights for each view.
- Experiments on seven datasets demonstrate robustness under high missing rates.

ARTICLE INFO

Keywords:
Incomplete multi-view clustering
Representation alignment
Representation fusion
Mutual information

ABSTRACT

Incomplete multi-view clustering (IMVC) aims to cluster data instances when some views are partially missing, which is a common challenge in real-world multi-modal scenarios. Existing IMVC methods often suffer from unreliable missing-view recovery, insufficient representation alignment, and the inability to adaptively exploit the varying importance of different views, leading to suboptimal clustering performance. To address these challenges, we propose an Adaptive Representation Learning framework for Incomplete Multi-View Clustering (ARLIMVC). The proposed framework integrates three key components: (1) a mutual information-guided data recovery module that adaptively weights inter-view relationships to improve missing-view reconstruction, (2) a dual-constraint representation alignment strategy that jointly enforces cross-view consistency and intra-view compactness in the latent space, and (3) a Kolmogorov–Arnold Networks (KAN)-based weighted fusion mechanism that dynamically captures the contribution of each view to the clustering task. These components are seamlessly embedded into a variational autoencoder architecture to enable end-to-end optimization. Extensive experiments on multiple benchmark datasets with varying missing rates demonstrate that ARLIMVC achieves superior or highly competitive performance in most cases in terms of clustering accuracy, robustness, and convergence stability. In particular, the proposed method achieves notable improvements in ACC, NMI, and ARI under high missing-rate scenarios, validating its effectiveness and practical applicability for incomplete multi-view clustering tasks.

☆ This work is supported by the Qinglan Project of Jiangsu Province of China, National Natural Science Foundation of China (No. 62006104), Jiangsu Normal University Faculty Research Excellence Development Program (No. JSNUGZL2026032), Postgraduate Research & Practice Innovation Program of Jiangsu Normal University (No. 2024XKT2629).

* Corresponding author.
Email address: dumj@jsnu.edu.cn (M. Du).

https://doi.org/10.1016/j.ins.2026.123588
Received 7 February 2025; Received in revised form 3 May 2026; Accepted 3 May 2026
Available online 4 May 2026

1. Introduction

Recently, investigations into multi-view observations have attracted extensive attention, because relying on a single view often inadequately captures the full characteristics of the original object [1]. Multi-view observations are typically gathered from different data sources or by different sensors, enabling a more thorough and holistic understanding of the subject. For example, doctors often combine information from different diagnostic imaging procedures, including computed tomography (CT) scans and magnetic resonance imaging (MRI), to form a more accurate diagnosis of cancer. Multi-view clustering, a fundamental challenge in multi-view learning, is an unsupervised task aimed at grouping unlabeled instance points by identifying shared clustering patterns across multi-view data, and has been extensively studied within the domains of machine learning and image analysis.

Current clustering methods for multi-view instances rely heavily on the assumption that the sample has complete views [2]. In practical applications, multi-view samples are often incomplete due to various factors, such as resource constraints, technical limitations, or equipment failures. For instance, medical data often include multiple views, such as imaging scans, genetic data, and medical records, but not all patients have all types of test results due to cost or necessity. Unfortunately, existing multi-view clustering methods often struggle to perform well with incomplete multi-view data. Therefore, an increasing number of scholars are now focusing on the challenge of incomplete multi-view clustering (IMVC).

To tackle the issue of incomplete multi-view clustering learning, a majority of IMVC methods commonly address missing views by imputing them before exploring clustering information [3]. Nevertheless, conventional IMVC techniques often exhibit limited capabilities in learning feature representations, while some of them involve high computational costs [4].

Over the past few years, remarkable advancements have been achieved in the IMVC field by integrating deep learning techniques [5]. These methods utilize the capabilities of deep learning models to perform data imputation and representation learning while incorporating clustering objectives into deep learning frameworks, yielding satisfactory clustering results [6]. Recent advances in incomplete multi-view clustering (IMVC) have witnessed a rapid expansion of methodological paradigms beyond traditional matrix factorization or autoencoder-based frameworks [7]. In particular, several representative research directions have emerged in recent years to address the challenges caused by severe view missingness and heterogeneous feature spaces [8]. Prototype-based IMVC methods have been extensively explored due to their intuitive interpretability and clustering-oriented design. These approaches aim to align view-specific representations by learning shared or matched prototypes across multiple views [9]. For example, PMIMC [10] exploits relational consistency and prototype-level contrastive learning to alleviate prototype misalignment under incomplete settings. Similarly, prototype matching learning further enforces cross-view prototype correspondence to enhance robustness against missing views. Despite their effectiveness, prototype-based strategies remain sensitive to heterogeneous architectures and high missing rates, where unreliable or noisy prototypes may lead to unstable optimization and degraded clustering performance. Another emerging research line focuses on generative modeling for IMVC, particularly diffusion-based frameworks. Diffusion-based methods [11] improve view recovery through denoising processes but often prioritize generative fidelity over clustering utility. However, diffusion-based methods emphasize generative fidelity and reconstruction quality, which are not always well aligned with clustering-oriented objectives, potentially leading to suboptimal representations for clustering. Beyond prototype and generative paradigms, recent studies have highlighted the importance of local reasoning and correlation-aware modeling in IMVC. DGIMVCM [12] captures fine-grained inter-view correlations and structural dependencies among samples through local neighborhood reasoning, thereby learning more discriminative representations from incomplete observations. Although these recent advances have significantly enriched the IMVC literature, most existing methods rely on either loosely coupled reconstruction and clustering objectives or rigid alignment strategies that lack adaptive weighting across views [13]. Moreover, insufficient coordination between missing-view recovery, representation alignment, and clustering optimization often limits their effectiveness in highly incomplete and heterogeneous multi-view scenarios.

However, despite these innovations, existing IMVC methods still face fundamental limitations in fully exploiting the potential of multi-view data. First, most methods focus solely on leveraging the existing views and overlook the recovery of missing views, which may lead to incomplete and biased representations. Second, traditional methods mainly focus on aligning representations across views using consistent information, but overlook the local distribution characteristics and inter-instance relationships, leading to fragmented and non-compact common representation spaces that hinder clustering performance. Third, neglecting the varying importance of views, where some views contribute more significantly to clustering while others are less influential, can result in undesirable feature alignment and fusion.

To tackle these issues, we propose an adaptive representation learning framework for incomplete multi-view clustering. Specifically, our framework consists of two main components: representation learning and clustering. The representation learning part mainly adopts the architecture of a variational autoencoder (VAE), containing three main modules: mutual information-guided data recovery module, dual constraint representation alignment module and weighted representation fusion module. The mutual information-guided data recovery module focuses on constructing a similarity-based relationship matrix and computing the mutual information between views to derive adaptive weights, which are used to guide the weighted recovery of missing data. Meanwhile, VAE is further employed to model complex inter-view relationships and ensure high-quality representations during data completion. The dual constraint representation alignment module, utilizing dual constraints, promotes feature consistency across different views by ensuring that latent representations derived from different views are aligned in a shared space. Finally, the weighted representation fusion module employs a Kolmogorov-Arnold networks (KAN)-based fusion strategy, which dynamically assigns importance to different views based on their contributions to the clustering task, allowing for the optimal integration of complementary information. The fused representation is passed into a clustering module to refine and generate the final clustering results.

Representatively, the primary contributions of this work can be summarized as follows:

- We propose an innovative framework that effectively integrates four key components: mutual information-guided data recovery, dual constraint representation alignment, weighted representation fusion, and clustering, to provide a unified solution for incomplete multi-view clustering problems.
- We propose a mutual information-guided reconstruction approach to recover missing data by adapting the data recovery process based on the relationships between available views, ensuring accurate and consistent data completion.
- We propose a weighted representation fusion approach that optimally combines the learned representations from different views, dynamically adjusting view weights based on their importance to the clustering task.

2. Related work

2.1. Deep multi-view clustering

Multi-view clustering methods utilize both consistent and complementary information contained across various views to deliver superior clustering outcomes. Unlike traditional shallow methods [14], deep multi-view clustering (DMVC) uses deep learning architectures to extract robust embeddings and capture complex non-linear relationships, making it well-suited for clustering high-dimensional or complex data.

A foundational approach in DMVC is the use of autoencoders (AEs) to extract view-specific latent representations while preserving the unique characteristics of each view. For instance, Yin et al. [15] propose a deep autoencoder-based multi-view clustering method that effectively disentangles unique and common information across views and fuses only the common parts to form a shared representation. However, such methods usually fail to fully utilize complementary information. To enhance information integration and utilize complementary information, Xu et al. [16] suggest using deep autoencoders to learn view-specific embeddings, which are then concatenated to create global features. This method effectively mitigates the challenges posed by unclear clustering structures by employing pseudo-labels to guide discriminative feature learning, thereby ensuring consistent cluster assignments across different views. Although previous methods show promise, representation alignment is rarely considered. Furthermore, Jin et al. [17] address the issue of view inconsistencies and non-aligned cluster prototypes by using pairwise data alignment as a proxy monitoring signal to guide correspondence construction across views. They also introduce a prototype alignment module to adjust and calibrate the distribution of incomplete data.

Despite these advancements, most existing DMVC methods operate under the assumption that all views are complete, limiting their applicability in real-world situations involving missing data.

2.2. Incomplete multi-view clustering

Incomplete multi-view clustering is alternatively referred to as partial multi-view clustering in existing research. Traditional IMVC methods use classical machine learning techniques such as matrix factorization, kernel functions, graph learning and tensor techniques. In short, matrix factorization-based incomplete multi-view clustering (MFIMC) aims to derive a unified low-dimensional representation for all views by applying matrix factorization methods. For instance, Li et al. [18] introduce an approach utilizing non-negative matrix factorization to manage incomplete multi-view data. Wen et al. [19] provide graph regularized matrix factorization, simultaneously utilizing both the local structure of individual views and the complementary relationships across views to derive a shared latent feature. Kernel-based techniques have gained significant attention for their capability to interpolate missing data by utilizing the kernel matrix derived from complete multi-view data. Liu et al. [20] propose the multiple kernel IMVC method, which fills in missing data in each view by using the learned consensus matrix. Guo et al. [21] develop an approach to address IMVC using core similarities and an anchor strategy. The graph-based IMVC (GIMVC) leverages the graph structure's information to enhance the identification of cluster patterns. In comparison with the methods based on matrix factorization, the GIMVC can make effective utilization of the geometric properties inherent in the data. Trivedi et al. [22] suggest filling in the missing instances by using the Laplacian matrix from complete views. Following this, they learn low-dimensional representations for different views using kernel canonical correlation analysis (CCA). Wen et al. [23] introduce a unified framework designed to simultaneously learn the completion graph and consensus representation by integrating intra-view and inter-view information. Recent work on tensor-based IMVC usually introduces low-rank tensor constraints to characterize the high-order relationships and intrinsic structures across multiple views. Hao et al. [24] combine low-rank data reconstruction with consistency guidance to enhance clustering performance by leveraging both consistent and complementary information across views.

Recently, deep learning-based IMVC has garnered growing interest [25]. Inference and imputation of missing instances are generally performed using the autoencoder, generative adversarial network, graph neural network, etc. Autoencoder derives representations by minimizing the discrepancy between the input data and its reconstructed output [26]. In multi-view clustering, deep autoencoder is often integrated with established clustering objectives to enhance performance [27]. Li et al. [28] leverage view-specific autoencoders to recover missing views while employing graph convolution and t-SNE regularization to extract unified, cluster-friendly representations from incomplete multi-view data. Unlike previous methods, which mainly rely on autoencoder or GAN, Pu et al. [29] integrate adaptive feature imputation using latent graphs and cross-view clustering information to enhance discriminative representation and clustering performance. Wang et al. [30] introduce a graph comparison learning-based approach for incomplete multi-view clustering, which concurrently reconstructs missing data and learns representations by leveraging cluster-level consistency across views. Xu et al. [31] introduce a deep IMVC framework that avoids imputation and fusion, independently learns embeddings for each view, mines cluster complementarity, and utilizes an EM-like approach to achieve clustering consistency and superior performance on incomplete multi-view data.

Although previous work has made a promising effort in deep incomplete multi-view clustering, these methods often overlook the differences in view contributions, treating all views equally.

3. Preparation

3.1. Notations

In this paper, we define the incomplete multi-view data including N instances with V views as $\boldsymbol{X} = \{\boldsymbol{X}^{(1)}, \ldots, \boldsymbol{X}^{(V)}\}$, and $\boldsymbol{X}^{(v)} = \{\boldsymbol{x}_1^{(v)}, \ldots, \boldsymbol{x}_N^{(v)}\} \in \mathbb{R}^{N \times D_v}$, where V denotes the total number of views, N is the number of samples and D_v signifies the feature dimensionality of v-th view. Missing instance points are marked as 'NaN'. In addition, we define the matrix $\boldsymbol{A} = \{0, 1\} \in \mathbb{R}^{N \times V}$ to track the availability and absence of these N samples, where $\boldsymbol{a}_i^{(v)} \in \boldsymbol{A}$, $\boldsymbol{a}_i^{(v)} = 0$ indicates that the i-th sample in the v-th view is missing, and $\boldsymbol{a}_i^{(v)} = 1$ signifies that the entry is present. The graph structure of the data is represented as an adjacency matrix $\boldsymbol{M} = \{\boldsymbol{M}^{(1)}, \ldots, \boldsymbol{M}^{(v)}, \ldots, \boldsymbol{M}^{(V)}\}$, where $\boldsymbol{M}^{(v)} \in \mathbb{R}^{N \times N}$. The element $M_{ij}^{(v)}$ indicates the presence or absence of an edge between data points $x_i^{(v)}$ and $x_j^{(v)}$. Furthermore, we denote K as the number of nearest neighbors used for relational graph construction and C as the number of clusters. Let D and H represent the average feature dimensionality and the average hidden-layer dimensionality across all views, respectively.

3.2. Variational autoencoder

To tackle the difficulties in incomplete multi-view clustering, this study integrates a variational autoencoder into the model architecture, replacing the traditional autoencoder. VAE incorporates principles from both variational inference and deep learning to create a continuous, lower-dimensional latent space, effectively enhancing its ability to capture complex inter-view relationships and compensate for missing data. Unlike AE, which directly maps inputs to deterministic latent representations, VAE aims to maximize the marginal likelihood $p_\theta(\boldsymbol{X})$, where $\boldsymbol{X}$ is the input data. By doing so, it provides a more robust mechanism for information fusion across multiple views.

The VAE consists of two primary components: a probabilistic encoder and a decoder. The encoder, parameterized by ϕ, approximates the true posterior distribution $p_\phi(\boldsymbol{z}|\boldsymbol{X})$, which is assumed to follow a Gaussian distribution. It outputs the mean $\boldsymbol{\mu}$ and log-variance $\log \boldsymbol{\sigma}^2$ of the latent distribution. The decoder, parameterized by θ, reconstructs the original input data $\boldsymbol{X}$ from the latent variable $\boldsymbol{z}$.

In our model, VAE serves as the backbone architecture of the proposed framework. Specifically, for each view v, the encoder maps the input $\boldsymbol{X}^{(v)}$ to a latent space defined by its mean $\boldsymbol{\mu}^{(v)}$ and log-variance $\log (\boldsymbol{\sigma}^{(v)})^2$. The process can be formulated as:

$$\left(\boldsymbol{\mu}^{(v)}, \log \boldsymbol{\sigma}^{(v)2}\right) = f_{\text{enc}}^{(v)}(\boldsymbol{X}^{(v)}) \tag{1}$$

where $f_{\text{enc}}^{(v)}(\cdot)$ represents the encoder network for the v-th view. Each encoder is composed of multiple fully connected layers, with ReLU activations ensuring non-linearity.

Using the reparameterization trick, the latent representation $\boldsymbol{z}^{(v)}$ is computed as:

$$\boldsymbol{z}^{(v)} = \boldsymbol{\mu}^{(v)} + \boldsymbol{\sigma}^{(v)} \odot \boldsymbol{\epsilon} \tag{2}$$

where $\odot$ denotes as a multiplication operation, and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. This ensures that the latent variable is differentiable with respect to the model parameters.

The decoder reconstructs the input $\boldsymbol{X}^{(v)}$ from the common representation $\boldsymbol{Z}^*$. The decoding process is expressed as:

$$\hat{\boldsymbol{X}}^{(v)} = f_{\text{dec}}^{(v)}(\boldsymbol{Z}^*) \tag{3}$$

where $f_{\text{dec}}^{(v)}(\cdot)$ denotes the decoder network for the v-th view. The fused latent variable $\boldsymbol{Z}^*$ is computed using a weighted representation fusion mechanism; see subsequent sections for specific descriptions.

To further enhance clustering performance, the introduction of a β-weighted KL divergence balances reconstruction fidelity and latent space regularization, providing a robust mechanism for information fusion across multiple views and yielding superior results in incomplete multi-view scenarios.

3.3. Kolmogorov–arnold networks

Recently, Liu et al. [32] introduced Kolmogorov-Arnold Networks, inspired by the Kolmogorov-Arnold representation theorem. Unlike traditional neural networks such as multilayer perceptrons (MLPs), which are based on the universal approximation theorem, KANs leverage this theorem to achieve high non-linear expressiveness with lower model complexity offering a new perspective for tackling complex problems.

The Kolmogorov-Arnold representation theorem states that any continuous multivariate function over a bounded domain can be decomposed into a sum of univariate functions. Formally, a function $f(\boldsymbol{x}) : [0, 1]^N \to \mathbb{R}$ can be expressed as:

$$f(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N) = \sum_{m=1}^{2N+1} \boldsymbol{\Phi}_m \left(\sum_{n=1}^{N} \boldsymbol{\phi}_{m,n}(\boldsymbol{x}_n) \right) \tag{4}$$

Here, $\boldsymbol{\phi}_{m,n}$ and $\boldsymbol{\Phi}_m$ are continuous univariate functions. Despite its mathematical elegance, this theorem has been overlooked in machine learning. However, Liu et al. [32] explore its potential for developing new neural network architectures, proposing the Kolmogorov-Arnold network, where all learnable components are univariate functions parameterized as B-splines. Unlike traditional networks, KAN treats activation functions as trainable components, allowing dynamic adaptation to complex data distributions. This enables the network to automatically adapt to the most appropriate non-linear transformations, offering enhanced flexibility in handling complex data distributions and intricate non-linear relationships. A KAN layer is defined by a matrix of univariate functions:

$$\boldsymbol{\Phi} = \{\boldsymbol{\phi}_{m,n}\}, \quad n = 1, \ldots, N_{\text{in}}, \quad m = 1, \ldots, N_{\text{out}} \tag{5}$$

where N_{in} and N_{out} represent the number of inputs and outputs, respectively, and each function $\boldsymbol{\phi}_{m,n}$ possesses trainable parameters. This allows the original Kolmogorov-Arnold representations—first defined as a two-layer composition—to be extended into more complex, deeper architectures. This structure generalizes the Kolmogorov-Arnold theorem into deeper architectures. For example, the original theorem corresponds to a two-layer KAN, while deeper models are defined as a composition of L layers:

$$\hat{f}(\boldsymbol{x}) = f_{KAN}(\boldsymbol{x}) = (\boldsymbol{\Phi}_L \circ \boldsymbol{\Phi}_{L-1} \circ \ldots \circ \boldsymbol{\Phi}_1)\boldsymbol{x} \tag{6}$$

where each $\boldsymbol{\Phi}_l$ denotes the function matrix corresponding to the l-th layer of the KAN, and $\circ$ denotes a composite mapping of $\boldsymbol{\Phi}_L$ and $\boldsymbol{\Phi}_{L-1}$. This deeper structure enhances the expressive power, enabling KAN to approximate more complex functions effectively.

3.4. Mutual information theory

Mutual information (MI) is commonly used to describe the correlation between two random variables. For two discrete random variables X and Y, if $(x, y) \sim p(x, y)$, the mutual information $I(X;Y)$ between X and Y can be defined as:

$$I(X;Y) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{7}$$

where $p(x, y)$ represents the joint probability distribution of X and Y, and $p(x) = \sum_y p(x, y)$ and $p(y) = \sum_x p(x, y)$ are the corresponding marginal probability distributions.

Mutual information quantifies the amount of shared information between the random variables X and Y, reflecting the degree to which they contain information about each other. When X and Y are independent, their joint distribution $p(x, y)$ is equal to the product of their marginal distributions, i.e., $p(x, y) = p(x)p(y)$. In this case, the mutual information $I(X;Y)$ equals zero.

4. Method

4.1. Overall model of ARLIMVC

The proposed ARLIMVC is designed to address the challenges of missing data, representation inconsistency, and unequal view contributions in a unified manner. Instead of treating data recovery, representation alignment, and multi-view fusion as independent stages, ARLIMVC tightly couples these components into a single end-to-end optimization framework, enabling robust and discriminative representation learning under incomplete multi-view scenarios.

As illustrated in Fig. 1, the framework consists of two main components: representation learning and clustering. The representation learning component is built upon a variational autoencoder architecture and integrates three key modules: mutual information-guided data recovery, dual-constraint representation alignment, and weighted representation fusion. These modules are designed to function collaboratively, with each module progressively enhancing the quality of the learned representations.

The mutual information-guided data recovery module serves as the foundation of the framework. Its primary function is to mitigate the adverse impact of missing views by reconstructing incomplete data in a structure-aware manner. Unlike conventional imputation methods that treat each view independently, this module exploits inter-view mutual information to estimate the reliability of different views and derive adaptive weights for data recovery. This design allows the model to preserve cross-view correlations while reducing the influence of unreliable or noisy views during the recovery process. The recovered features are subsequently encoded by the VAE to obtain informative latent representations.

Based on the recovered representations, the dual-constraint representation alignment module aims to learn a shared latent space that is both consistent across views and compact within each view. Specifically, a cross-view consistency constraint encourages latent representations of the same instance from different views to be aligned in the shared space, while an intra-view constraint preserves local neighborhood structures and enhances intra-class compactness. By jointly enforcing these complementary constraints, this module alleviates representation fragmentation and provides a stable latent space for effective multi-view fusion.

Finally, the weighted representation fusion module integrates the aligned representations from multiple views into a unified representation for clustering. Motivated by the observation that different views contribute unequally to clustering performance, this module adopts a KAN-based fusion strategy to dynamically learn the importance of each view in a data-driven manner. Benefiting from the aligned latent space, the fusion module focuses on complementary information while suppressing redundant or noisy features. The fused representation is then fed into the clustering component to generate refined clustering results.

Overall, the proposed ARLIMVC framework forms a coherent pipeline in which data recovery reduces incompleteness-induced bias, representation alignment ensures structural consistency, and adaptive fusion effectively integrates multi-view information. This collaborative design enables ARLIMVC to learn robust and discriminative representations for incomplete multi-view clustering.

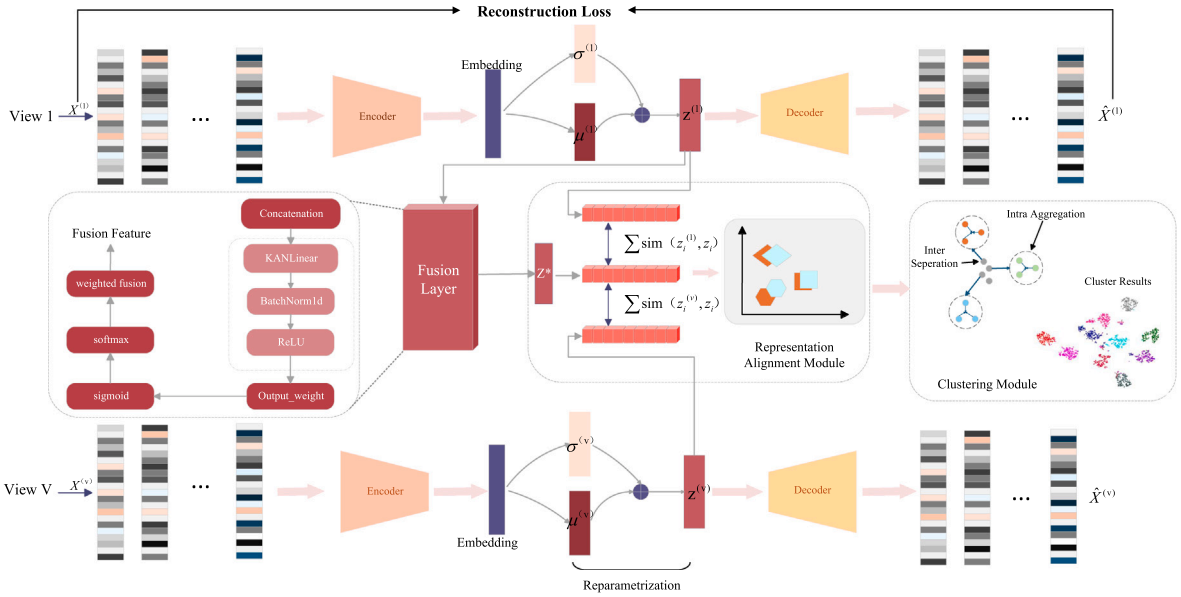


Fig. 1. The framework of our model. It consists of four main components, i.e., latent representation learning, representation fusion, representation alignment and clustering.

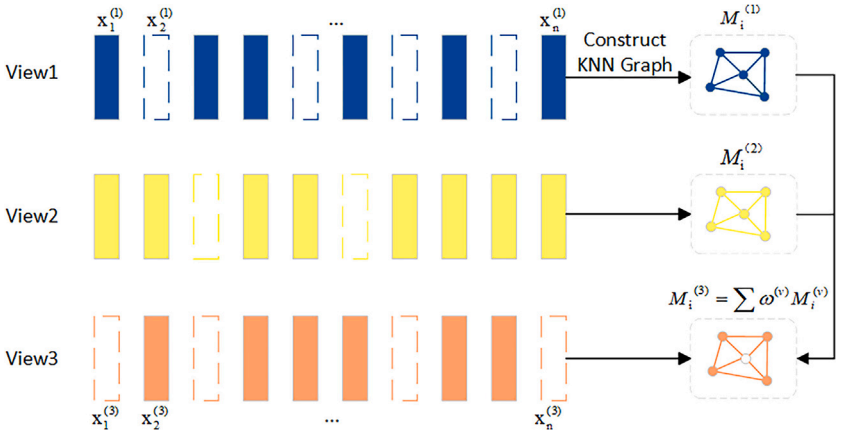


Fig. 2. Description of constructing relational graphs on missing views and using weighted completions.

4.2. Mutual information-guided data recovery module

To address the issue of missing data, we hypothesize that similar samples, especially those belonging to the same cluster, should exhibit similar representations across different views. Moreover, each view contributes differently to the reconstruction of missing data; that is, certain views may offer more pertinent information for specific instances of missing data. To achieve this, we propose the construction of an adaptive relational graph based on the nearest neighbors, which identifies similar samples in the complete views. This graph is then transferred to the missing views using a weight derived from the mutual information between views.

As shown in Fig. 2, we illustrate the scenario of missing views in multi-view data using three views, and show how to build the relational graph for the missing information. The blanks in the figure indicate the missing view. For instance, in Fig. 2, the third view of the latest instance is missing, and the corresponding first and second views are available. By transferring the relational graph constructed from the first and second views to the third view using weights, we identify samples that are similar to the missing data with greater precision.

For reconstructing a missing view, we use an adaptive approach that integrates information from the available views through the construction of a weighted relational graph. The adaptive weighted relational graph is formulated as follows:

$$\boldsymbol{M}_i^{(j)} = \sum_{v} \boldsymbol{w}^{(v)} \boldsymbol{M}_i^{(v)}, \quad j \neq v \tag{8}$$

where $\boldsymbol{M}_i^{(v)}$ is the i-th row of the relational graph from the v-th view, and $\boldsymbol{w}^{(v)}$ is the mutual information-derived weight for view v. This approach ensures that the relational graph for missing views accurately reflects the relationships in the existing views, effectively utilizing complementary information.

We first introduce methods for constructing graphs on partial multi-view data. We compute the similarity between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in the v-th view.

Then, we construct the relational graph $\boldsymbol{M}$ using K-nearest neighbors based on similarity. The relational graph is constructed by identifying the nearest neighbors for each sample, where $\boldsymbol{M}_i^{(v)}$ indicates the similarity between the i-th sample and its neighbors in the v-th view, which captures the structure of the data within that particular perspective. By combining these relational graphs from all available views, we can leverage the full spectrum of information to ensure that the reconstructed relational structure for the missing view is both accurate and reflective of the overall data distribution.

To consider the importance of different views, we compute the mutual information between the feature representations of different views. This mutual information is used to assign weights to each view, indicating its relative contribution to capturing sample relationships.

The feasibility and rationality of this approach lie in the fact that the greater the mutual information between the feature representations $\boldsymbol{Z}^{(v)}$ of different views in a multi-view data representation, the higher the correlation between the views. This implies that the importance of the view is greater, and thus, its corresponding weight should also be larger. Moreover, since mutual information does not require the introduction of additional parameters or assumptions, this measurement method is highly suitable for complex models such as multi-view clustering. The weight solution for each view can be defined as

$$\boldsymbol{w}^{(v)} = \frac{I(\boldsymbol{Z}^{(v)}; \boldsymbol{Z}^{(j)})}{\sum_j I(\boldsymbol{Z}^{(v)}; \boldsymbol{Z}^{(j)})} \tag{9}$$

4.3. Dual-constraint representation alignment module

After processing the multi-view missing values, we obtain complete multi-view representations. However, these completed representations often fail to ensure that representations from different views are unified into a shared common representation space, which is crucial for extracting consistent and discriminative features for clustering tasks. Therefore, it is necessary to refine the representations generated by the multi-view missing value processing module by aligning features across views. Traditional representation alignment schemes often use the Euclidean norm to align representations of the same instance in different views. However, these methods ignore the local distribution characteristics of the views. In this regard, we use the distance constraint between nearest neighbors to align representations across different views. Furthermore, traditional representation alignment schemes primarily aim to ensure similarity between representations of the same instance across different views, while neglecting the potential relationships between the instance and other similar instances. This results in a less compact distribution in the common representation space, thereby reducing the effectiveness of clustering. To this end, we use a neighbor-level contrastive learning approach to refine the representations, ensuring that instances within the same view exhibit greater similarity to their neighbors.

In detail, we use a representation alignment strategy with dual constraints. We apply a distance-based cross-view local constraint to align representations across different views within the common representation space, and a contrastive learning-based intra-view local constraint to improve the compactness of the distribution in this space. More specifically, rather than focusing solely on the distance constraints between pairs of instances from different views, we emphasize the distance constraints between their nearest neighbors.

For each instance in a given view, we first obtain its set of K-nearest neighbors based on the previously constructed relational graph. These neighbors capture the local structure and semantic relationships surrounding the instance. Then, we incorporate KNN information into a distance-based constraint to align representations across views within the common representation space, as described below:

$$\mathcal{L}_{k_i}^{(v)} = \frac{1}{K} \sum_{j \neq v}^{V} \sum_{k=1}^{K} \left\| \boldsymbol{z}_{i^k}^{(v)} - \boldsymbol{z}_{i^k}^{(j)} \right\|_2^2 \tag{10}$$

where $\boldsymbol{z}_{i^k}^{(v)}$ is the k-th nearest neighbor of $\boldsymbol{z}_i^{(v)}$, which is obtained from the previously constructed relational graph and, consequently, does not require re-computation. Extending Eq. (10) from $\boldsymbol{z}_i^{(v)}$ to the whole we can obtain the distance-based cross-view local constraint loss as follows.

$$\mathcal{L}_{KNN} = \frac{1}{N \cdot V} \sum_{i=1}^{N} \sum_{v=1}^{V} \mathcal{L}_{k_i}^{(v)} \tag{11}$$

On the other hand, we use neighbor-level contrastive learning to achieve an intra-view local constraint, which is used to enhance the compactness of the distribution of the common representation space. The core idea behind contrastive learning is to maximize the similarity of positive pairs while minimizing the similarity of negative pairs. In our method, we identify the most similar instances within the same view as positive pairs. Based on this foundational principle, the developed representation alignment module can further improve the quality of the learned representations, thereby enhancing clustering performance.

For each representation $z^{(v)}$, we can obtain K positive pairs depending on whether the sample belongs to $M_i^{(v)}$. To measure the similarity between the two representations, we use cosine distance. The specific definition of the formula is as follows:

$$s\left(z_i^{(v)}, z_j^{(v)}\right) = \frac{\left(z_i^{(v)}\right)^T \left(z_j^{(v)}\right)}{\left\|z_i^{(v)}\right\| \left\|z_j^{(v)}\right\|}. \tag{12}$$

where $z_i^{(v)}$ and $z_j^{(v)}$ represent two representation vectors, and $\|\cdot\|$ denotes the Euclidean norm. Positive pairs are encouraged to exhibit high cosine similarity, while negative pairs are pushed apart to preserve feature distinctiveness. For each representation $z_i^{(v)}$ the contrastive loss is defined as:

$$\mathcal{L}z_i^{(v)} = -\frac{1}{K}\sum_{k=1}^{K} \log \frac{\exp\left(s\left(z_i^{(v)}, z_{i^k}^{(v)}\right)\right)}{\sum_{n=1}^{N}\left[\exp\left(s\left(z_i^{(v)}, z_n^{(v)}\right)\right) + \exp\left(s\left(z_i^{(v)}, z_{n^k}^{(v)}\right)\right)\right]}, \tag{13}$$

where $z_{i^k}^{(v)}$ is one of the nearest neighbors related to $z_i^{(v)}$, and $z_n^{(v)}$ represents other features in the same mini-batch. Similarly, $z_{n^k}^{(v)}$ corresponds to one of the nearest neighbors related to $z_n^{(v)}$.

To extend this loss to all views and samples, we define the total contrastive learning-based intra-view local constraint loss as:

$$\mathcal{L}_{CON} = \frac{1}{N \cdot V}\sum_{i=1}^{N}\sum_{v=1}^{V} \mathcal{L}z_i^{(v)} \tag{14}$$

where N is the number of samples, and V is the number of views.

Thus, the overall alignment loss formula can be described as:

$$\mathcal{L}_{ALI} = \mathcal{L}_{KNN} + \mathcal{L}_{CON} \tag{15}$$

By using the representation alignment with dual constraints, our method effectively transforms the data into a multi-view aligned space characterized by a compact distribution.

The proposed alignment loss is designed to jointly enforce cross-view consistency and intra-view compactness in the latent representation space. Specifically, the cross-view consistency constraint encourages representations of the same instance from different views to be aligned, which reduces view-specific discrepancies and promotes a shared latent structure. Meanwhile, the intra-view compactness constraint pulls samples with similar semantics closer together within each view, leading to more discriminative and clustering-friendly representations. Theoretically, these two constraints are complementary rather than conflicting. The cross-view alignment operates at the instance level across views, while the intra-view compactness focuses on local neighborhood structures within each view. Their joint optimization guides the latent representations toward a well-structured manifold where inter-view agreement and intra-view discrimination are simultaneously preserved.

From an optimization perspective, the alignment loss is formulated as a smooth and differentiable objective and is optimized jointly with the reconstruction loss of the variational autoencoder. This unified optimization framework enables stable gradient updates and avoids oscillatory behavior during training. Moreover, the smoothness of the alignment loss implicitly constrains gradient variations, which is closely related to the notion of Lipschitz continuity and gradient smoothing commonly adopted to improve optimization stability in deep models.

4.4. Weighted representation fusion module

The goal of the multi-view representation fusion approach is to integrate inputs from multiple views into a compact single representation. This approach is particularly important when dealing with multi-view data, as different views often contain complementary information that can provide a more comprehensive understanding. Traditional approaches, such as concatenating or averaging features from different views, often fail to capture the complementary information effectively, leading to suboptimal clustering performance. To address this limitation, in this module, inspired by Liu et al. [25], we propose an adaptive weighted mechanism for deep fusion, which dynamically aggregates features from multiple views by learning instance-specific weights. This mechanism ensures that more informative features are emphasized while less relevant features are attenuated, enabling effective utilization of the rich complementary information inherent in multi-view data. The fused multi-view representation Z^* is calculated as follows:

$$Z^* = \sum_{v=1}^{V} e^{(v)} \odot z^{(v)} \tag{16}$$

Here, $\odot$ represents element-wise multiplication, and $e = [e^{(1)}, \ldots, e^{(V)}] \in \mathbb{R}^{N \times Dv}$ contains the learned adaptive weights for view v, where $\sum_{i=1}^{V} e^{(i)} = 1$. The input representations $z^{(1)}, z^{(2)}, \ldots, z^{(V)}$ from all views are concatenated into a unified vector:

$$\widetilde{z} = [z^{(1)}, z^{(2)}, \ldots, z^{(V)}] \tag{17}$$

This concatenated feature representation is processed by KAN, which learns the complex relationships between views and generates a fused intermediate representation:

$$\widetilde{G} = f_{KAN}(\widetilde{z}) \tag{18}$$

Specifically, firstly, the shared representation undergoes a nonlinear transformation using the $SiLU$ activation function, followed by a linear transformation with the weight matrix $\boldsymbol{W}_b$ to produce the base output. Concurrently, B-spline basis functions are recursively computed for each input feature $\boldsymbol{z}_i$, resulting in a matrix function $\boldsymbol{B}(\widetilde{\boldsymbol{z}})$, which is then linearly transformed with the weight matrix $\boldsymbol{W}_s$ to generate the spline output. The final output $\boldsymbol{Y}$ is obtained by combining the base output and the spline output:

$$\boldsymbol{Y} = f_{SiLU}(\widetilde{\boldsymbol{z}})\boldsymbol{W}_{\mathrm{b}} + \boldsymbol{B}(\widetilde{\boldsymbol{z}})\boldsymbol{W}_{\mathrm{s}} \tag{19}$$

These representations are then normalized via a sigmoid function and further refined by a softmax operation to produce a weight coefficient $\boldsymbol{e}^{(v)} \in \mathbb{R}^{N\times 1}$ which is calculated as follows:

$$[\boldsymbol{e}^{(1)}, \ldots, \boldsymbol{e}^{(v)}, \ldots, \boldsymbol{e}^{(V)}] = f_{softmax}(f_{sigmoid}(\widetilde{\boldsymbol{G}})/\tau) \tag{20}$$

In this equation, applying a sigmoid function before the softmax operation helps to prevent any single view from dominating with a weight close to one. By employing a weighted representation fusion module, the complementary aspects of the multi-view observations are effectively harnessed.

The weighted representation fusion mechanism ensures that the fusion process dynamically adjusts to the needs of each instance, effectively harnessing both complementary and shared information from all views. By incorporating nonlinear transformations and B-spline interpolation, the fused representation becomes semantically rich, compact, and expressive. This approach significantly enhances the clustering performance by aligning and combining multi-view features in a principled, data-driven manner.

4.5. Clustering module

In the training stage, we obtain the common representation $\boldsymbol{Z}^*$ through the weighted representation fusion module, where $\boldsymbol{z}_i^*$ denotes the latent representation of the i-th sample. Following previous deep clustering methods [30], we further fine-tune the network by minimizing a clustering loss based on KL divergence. The clustering loss is defined as:

$$\mathcal{L}_{CLU} = KL(\boldsymbol{P}||\boldsymbol{Q}) = \sum_{i=1}^{N}\sum_{j=1}^{C} \boldsymbol{p}_{ij}\log\frac{\boldsymbol{p}_{ij}}{\boldsymbol{q}_{ij}} \tag{21}$$

where $\boldsymbol{q}_{ij}$ is computed using the student's t-distribution as:

$$\boldsymbol{q}_{ij} = \frac{(1 + \|\boldsymbol{z}_i^* - \boldsymbol{\mu}_j\|^2)^{-1}}{\sum_{c=1}^{C}(1 + \|\boldsymbol{z}_i^* - \boldsymbol{\mu}_c\|^2)^{-1}}, \tag{22}$$

The cluster center $\boldsymbol{\mu}_j$ is initially determined using the k-means algorithm on the common representation $\boldsymbol{Z}^*$. The value $\boldsymbol{q}_{ij}$ represents the probability that the node $\boldsymbol{Z}^*$ is assigned to cluster j. The matrix $\boldsymbol{Q} = [\boldsymbol{q}_{ij}]$ shows the distribution of all node assignments to the different clusters. The target distribution, $\boldsymbol{P}$, is derived from $\boldsymbol{Q}$ and can be calculated as follows:

$$\boldsymbol{p}_{ij} = \frac{\boldsymbol{q}_{ij}^2 / \sum_{i=1}^{N} \boldsymbol{q}_{ij}}{\sum_{c=1}^{C}(\boldsymbol{q}_{ic}^2 / \sum_{i=1}^{N} \boldsymbol{q}_{ic})} \tag{23}$$

By minimizing the KL divergence between $\boldsymbol{P}$ and $\boldsymbol{Q}$, we obtain a more suitable representation for clustering, with higher-confidence predictions. This process leads to the final consistent representation $\boldsymbol{Z}^*$ and the corresponding clustering assignments $\boldsymbol{Q}$.

4.6. Objective function and optimization process

The loss function for the VAE combines reconstruction loss and a KL divergence term. To balance reconstruction accuracy and latent space regularization, we introduce β in our implementation. For each view v, the VAE loss is formulated as:

$$\mathcal{L}_{VAE}^{(v)} = \mathcal{L}_{REC}^{(v)} + \beta\mathcal{L}_{KL}^{(v)} \tag{24}$$

where the reconstruction term $\mathcal{L}_{REC}^{(v)}$ measures the discrepancy between the original input $\boldsymbol{x}^{(v)}$ and its reconstruction $\hat{\boldsymbol{x}}^{(v)}$. To enhance robustness, we use a weighted mean squared error (WMSE):

$$\mathcal{L}_{REC}^{(v)} = \frac{1}{n}\sum_{i=1}^{n} \|\boldsymbol{x}_i^{(v)} - \hat{\boldsymbol{x}}_i^{(v)}\|^2 \tag{25}$$

The KL divergence regularizes the latent space by constraining the approximate posterior $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ to be close to the prior $p(\boldsymbol{z})$. It is computed as:

$$\mathcal{L}_{KL}^{(v)} = -\frac{1}{2}\sum_{j=1}^{V}\left(1 + \log\boldsymbol{\sigma}_j^2 - \boldsymbol{\mu}_j^2 - \boldsymbol{\sigma}_j^2\right) \tag{26}$$

The total loss aggregates contributions from all views:

$$\mathcal{L}_R = \sum_{v} \mathcal{L}_{VAE}^{(v)} \tag{27}$$

Based on the above description, we design the overall objective function of the proposed method as follows:

$$\mathcal{L} = \mathcal{L}_R + \alpha \mathcal{L}_{ALI} \tag{28}$$

where $\mathcal{L}_R$ and $\mathcal{L}_{ALI}$ are reconstruction loss and alignment loss, respectively. α is the trade-off hyperparameter for representation alignment. The proposed ARLIMVC is optimized within a unified framework using a stage-wise training strategy. In the first stage, the model focuses on representation learning by minimizing the VAE-based reconstruction loss together with the dual-constraint representation alignment loss, which enables robust latent representations to be learned from incomplete multi-view data. After the representation learning stage converges, a clustering loss is introduced to fine-tune the shared latent representation, encouraging a more discriminative and clustering-friendly embedding.

Algorithm 1 summarizes the optimization process.

Algorithm 1 Optimization of our proposed method.

Input: Dataset $\boldsymbol{X} = \{\boldsymbol{X}^{(1)}, \ldots, \boldsymbol{X}^{(V)}\}$, numbers of cluster $\boldsymbol{C}$, hyper-parameter K, total iteration numbers *epochs*
Output: Clustering predictions

```
Calculate the adjacency matrix M^(v) and the mutual information-based inter-view weights w^(v)
Transfer weighted adjacency matrix to missing instances M_i^(j) = Σ_v w^(v) M_i^(v)
for epoch = 1 → epochs do
    Learn view features
    μ^(v), log(σ^(v))^2 = f_enc^(v)(x^(v))
    z^(v) = μ^(v) + σ^(v) ⊙ ε,   ε ~ N(0, I)
    Compute the fused representations G̃ by Eq. (18)
    Calculate the loss L_ALI by Eq. (15)
    Calculate the loss L_R by Eq. (27)
    Calculate the overall loss L by Eq. (28)
end for
Calculate the clustering loss L_CLU by Eq. (21)
```

4.7. Time complexity analysis

The computational complexity of the ARLIMVC framework is primarily determined by its four collaborative modules. Let N denote the number of samples, V the number of views, K the number of neighbors, d the latent dimension, and C the number of clusters. We denote D and H as the average feature and hidden layer dimensionalities across all views, respectively.

The mutual information-guided data recovery module involves KNN graph construction and adaptive weight estimation across views, resulting in a complexity of $O(VNKD + NV^2)$. The representation learning module employs VAE-based encoders and decoders for all views, whose computational cost scales as $O(VNDH)$. The dual-constraint alignment module enforces cross-view local consistency and intra-view contrastive alignment, leading to a complexity of $O(NV^2\ Kd)$ for neighbor-level representation alignment. The KAN-based adaptive weighted fusion module incurs a cost of $O(NVd)$, and the KL-divergence-based clustering module contributes $O(NCd)$. Overall, the per-iteration computational complexity of ARLIMVC is dominated by representation learning and dual-constraint alignment, which can be summarized as $O(VNDH + NV^2\ Kd)$. This complexity is computationally feasible for practical incomplete multi-view clustering tasks.

To further demonstrate the computational efficiency of ARLIMVC, we compare its complexity with several representative IMVC methods under the same per-iteration setting. DAIMC is based on a weighted semi-NMF framework with alternating optimization, whose dominant computational cost is $O(VD^3 + NVDC)$. IMVC-IE integrates data inference and deep evaluation modules, where the autoencoder-based reconstruction and feature-level contrastive learning lead to a complexity of $O(VNDH + V^2N^2)$. DIMVC adopts an EM-like optimization strategy without explicit imputation or fusion, resulting in a total complexity of $O(C^3 + NDC + NVH^2)$. PMIMC employs view-specific reconstruction and relational consistency learning, and its overall complexity is dominated by deep network training and pairwise relational modeling, yielding $O(N^2\ VLH^2)$, where L is the network depth.

Compared with these methods, ARLIMVC avoids expensive matrix factorization, quadratic sample-wise contrastive operations, and high-order tensor decomposition. Although ARLIMVC introduces additional alignment and adaptive fusion components, its asymptotic computational complexity remains comparable to existing deep IMVC methods, while achieving improved representation quality and clustering robustness.

5. Experiments

5.1. Experimental settings

5.1.1. Datasets and metrics

The following seven datasets are used to assess the proposed method.

COIL20: The COIL20 dataset consists of 20 objects, each captured from 72 different angles, making up a total of 1440 grayscale images. It is widely used for image recognition tasks due to its consistent viewpoints and object variations.

Table 1
Statistics of multi-view datasets for experiments, where N, V, C, and Dv denote the number of objects, views, groups, and attributes, respectively.

Dataset	N	V	C	Dv
COIL20	1440	3	20	1024/1024/324
handwritten	2000	5	10	240/76/216/47/64
100leaves	1600	3	100	64/64/64
ORL	400	4	40	521/59/864/254
MSRC	210	5	7	24/576/512/256/254
Scene-15	4485	3	15	20/50/59
ALOI-100	10,800	4	100	77/13/64/125

100leaves: The 100leaves dataset contains 1600 images of leaves from 100 plant species. Each species has 16 images that vary in shape and texture.

Handwritten: This dataset contains images of handwritten digits, with 2000 samples for each of the 10 digits (0-9).

MSRC: The MSRC dataset is composed of 240 images across 8 object categories, such as animals, vehicles, and buildings.

ORL: The ORL dataset consists of 400 facial images from 40 individuals, with 10 different images per person. The images include variations in facial expressions and lighting conditions, making this dataset suitable for facial recognition studies.

Scene − 15: The Scene-15 dataset comprises 4485 samples across 15 scene categories, with each sample represented by three visual features: GIST, PHOG, and LBP.

ALOI − 100: The ALOI-100 dataset consists of 10,800 images, evenly distributed among 100 object classes. For multi-view learning, we characterize each image by extracting four complementary feature types: HSB, RGB, Colorsim, and Haralick.

The characteristics of the seven datasets are outlined in Table 1. Here, N represents the total number of instances, V indicates the number of views, and C refers to the number of clusters. Additionally, Dv signifies the number of features in the v-th view.

To assess performance, we use two common clustering metrics: accuracy (ACC) and normalized mutual information (NMI). The larger these metrics are, the better the clustering performance.

5.1.2. Baselines

In order to verify the validity and superiority of the method in this paper, two commonly used single-view clustering methods and seven latest IMVC methods are used to compare with the method proposed in this paper. These methods are listed below:

BSV [33]: This method applies the k-means clustering algorithm separately to each view and selects the best result as the final output.

Concat [33]: In this approach, all views are combined into a single dataset, followed by applying k-means clustering to obtain the final result.

PVC [15]: This technique uses non-negative matrix factorization (NMF) to uncover latent representations of incomplete multi-view data within subspaces.

DIMVC [31]: This is an imputation-free and fusion-free method for incomplete multi-view clustering, which works by mining the complementary information among clusters.

CDIMC − net [34]: By integrating view-specific deep encoders and a graph embedding strategy, it captures both high-level features and local structures of each view.

IMVC − IE [35]: Mining potential information through comparative learning of multi-view features and combining innovative evaluation strategies to infer and select appropriate data to complete the missing view, thus effectively addressing the deficiencies of existing methods in model depth and recovery data quality evaluation.

DAIMC [36]: reduces the impact of view incompleteness on clustering by enforcing alignment of the basis matrices of individual views with the help of regression.

APADC [37]: uses an imputation-free approach, learning features by self-encoder, avoiding filling of missing data by using adaptive feature projection, and combining mutual information maximization with distribution alignment optimization.

PMIMC [10]: A prototype matching learning method that addresses prototype misalignment and performance instability in incomplete multi-view clustering.

5.1.3. Implementation details

The proposed method leverages view-specific variational autoencoders, each built using fully connected (Fc) networks. For each view $\boldsymbol{X}^{(v)}$, the encoder follows the architecture $\boldsymbol{X}^{(v)} - Fc1024 - Fc1024 - Fc1024 - (\boldsymbol{\mu}^{(v)}, \boldsymbol{\sigma}^{(v)})$, where $\boldsymbol{\mu}^{(v)}$ and $\boldsymbol{\sigma}^{(v)}$ represent the learned mean and standard deviation for the latent distribution. The decoder is structured as $\boldsymbol{z}^{(v)} - Fc1024 - Fc1024 - Fc1024 - \hat{\boldsymbol{X}}^{(v)}$, where $\hat{\boldsymbol{X}}^{(v)}$ represents the reconstructed data. Each fully connected layer (Fc1024) contains 1024 neurons, with batch normalization and ReLU activation applied to all layers. The latent variable $\boldsymbol{z}^{(v)}$ is sampled from $\mathcal{N}(\boldsymbol{\mu}^{(v)}, \mathrm{diag}(\boldsymbol{\sigma}^{(v)})^2)$ using the reparameterization trick. The reconstruction loss and KL divergence are combined to optimize the VAE, ensuring both accurate data reconstruction and well-regularized latent representations.

In terms of dimensionality, unlike other methods that optimize representation dimensions within a range (e.g., 16 to 128), we fix the dimensionality of the learned latent representations to match the number of clusters C. This approach simplifies the model

Table 2
Clustering on seven multi-view datasets with different retained-view rates ($r \in \{50\%, 70\%, 90\%\}$).

r	Method	COIL20		100Leaves		HW		ORL		MSRC		Scene-15		ALOI-100	
		ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
50%	Concat	27.86	41.55	26.19	55	25.85	21.54	25.46	43.69	46.84	32.05	23.95	24.1	20.15	37.92
	PVC	51.48	59.31	19.45	48.54	46.82	37.31	27.93	46.45	44.29	32.42	25.61	25.31	24.84	40.15
	BSV	39.78	46.23	31.07	50.45	44.16	41.8	29.74	47.43	55.04	44.37	23.71	22.48	21.63	36.54
	DIMVC	69.14	66.79	50.6	68.98	79.82	66.43	57.38	76.89	65.28	54.36	33.62	30.01	55.15	69.71
	CDIMC-net	73.25	82.57	25.2	56.97	87.37	80.31	47.85	52.73	61.56	49.32	30.14	29.48	56.73	68.02
	APADC	53.4	61.9	38.03	61.58	64.43	56.42	49.54	66.48	48.71	40.97	38.89	35.4	29.57	55.14
	IMVC-IE	57.53	61.72	25.64	55.75	68.63	60.75	40.67	51.38	53.68	41.74	30.96	29.89	29.74	55.76
	DAIMC	67.04	73.75	24.74	56.25	78.46	64.37	54.81	74.15	59.83	56.74	23.6	21.88	58.82	70.46
	PMIMC	68.33	79.01	65.24	80.63	90.85	83.74	44.57	67.36	64.67	56.52	39.2	35.51	71.18	82.16
	ARLIMVC	**82.92**	**88.99**	**69.25**	**82.61**	**91.4**	**86.39**	**68.25**	**82.65**	**66.24**	**58.28**	**39.46**	**36.8**	**72.44**	**84.21**
70%	Concat	33.42	45.89	33.88	62.46	32.99	29.9	35.84	50.26	61.39	49.24	26.78	25.06	23.04	42.83
	PVC	57.77	68.13	20.74	51.1	55.14	46.88	39.02	52.43	50.65	44.79	29.51	27.9	28.79	47.83
	BSV	51.97	58.59	42.8	62.56	55.76	52.09	43.79	52.27	60.56	50.34	26.47	23.54	23.87	39.62
	DIMVC	71.59	70.36	62.86	77.24	83.72	74.78	62.14	80.26	70.43	59.38	33.71	32.53	59.7	74.13
	CDIMC-net	75.18	83.28	30.29	62.32	89.45	83.23	51.83	68.28	65.43	54.78	30.18	30.57	62.84	74.31
	APADC	61.5	70.2	50.77	76.07	75.26	67.59	56.32	71.46	52.94	49.96	39.99	40.63	34.54	58.11
	IMVC-IE	63.26	68.64	30.02	61.35	74.32	73.9	49.26	58.97	60.54	49.53	34.26	30.87	35.81	60.3
	DAIMC	70.8	78.67	34.17	63.41	82.83	73.92	58.39	79.68	68.68	63.45	26.49	22.25	65.84	76.91
	PMIMC	68.96	81.23	79.69	91.93	91.95	85.8	46.75	70.26	71.82	63.17	39.71	**40.21**	75.92	**87.35**
	ARLIMVC	**84.51**	**91.15**	**83.44**	**92.61**	**93.5**	**87.21**	**83.25**	**92.05**	**73.81**	**64.93**	**41.01**	40.17	**78.46**	86.04
90%	Concat	54.58	63.67	50.71	74.52	54.99	49.82	42.29	55.82	65.43	54.27	28.93	28.11	27.93	47.82
	PVC	62	74.66	21.02	53.1	63.74	58.67	47.34	59.09	67.55	60.24	30.83	31.05	32.73	53.82
	BSV	65.46	71.99	55	74.95	70.87	66.65	50.83	64.75	62.33	52.45	26.88	25.38	26.84	46.89
	DIMVC	73.69	72.55	71.97	85.94	84.28	76.85	66.54	84.35	75.25	67.44	34.15	33.26	67.58	81.26
	CDIMC-net	76.09	84.68	63.94	83.86	92.32	87.48	59.86	75.73	71.84	63.49	31.57	32.53	68.93	79.94
	APADC	70.9	81.6	66.48	85.09	88.23	81.73	66.73	85.65	66.65	59.07	41.98	40.05	35.81	59.69
	IMVC-IE	70.45	76.53	51.64	73.65	81.82	77.53	57.94	70.57	71.62	58.53	40.15	39.39	38.91	65.52
	DAIMC	71.61	80.09	59.56	79.86	83.81	75.72	64.02	82.59	75.35	66.74	32.09	33.55	70.43	81.96
	PMIMC	72.22	82.32	83.81	93.45	92.7	85.82	47.75	72.41	74.29	63.74	41.97	40.71	74.38	86.39
	ARLIMVC	**85.23**	**92.07**	**89.25**	**96.11**	**95.2**	**89.95**	**84.01**	**92.73**	**75.71**	**67.77**	**42.97**	**41.25**	**79.79**	**90.56**

by eliminating the need for dimensionality search while ensuring that the learned features capture cluster characteristics rather than individual view-specific traits. The encoder and decoder architectures in the VAE are defined by layer sizes $[0.8Dv, 0.8Dv, 1500, C]$ and $[C, 1500, 0.8Dv, 0.8Dv, Dv]$, respectively, where Dv is the original feature dimension of the v-th view. The encoder further produces the mean $\boldsymbol{\mu}^{(v)}$ and standard deviation $\boldsymbol{\sigma}^{(v)}$ for the latent variable $\boldsymbol{z}^{(v)}$, sampled using the reparameterization trick from $\mathcal{N}(\boldsymbol{\mu}^{(v)}, \mathrm{diag}(\boldsymbol{\sigma}^{(v)})^2)$, to optimize both reconstruction accuracy and latent space regularization.

We evaluate the performance of the comparison methods using open-source implementations. The experiments are conducted on a Windows machine with an NVIDIA GeForce RTX 4060 GPU (16GB memory) and an 11th Gen Intel Core i5-11600KF @ 3.90GHz processor. The code is available at: https://github.com/UPON1214/ARLIMVC.

5.2. Performance comparisons and analysis

To validate the overall performance of our method, we compare it with baseline approaches such as BSV, Concat, PVC, DIMVC, CDIMC-net, IMVC-IE, DAIMC, and APADC. Given that these methods employ different strategies for determining thresholds, we apply a uniform strategy for each possible threshold during the model's testing phase to ensure a fair comparison.

To ensure a fair and transparent comparison, all baseline results reported in Table 2 are obtained by re-running the publicly available implementations under a unified experimental protocol, rather than directly copying the values from the original papers. Specifically, all methods are evaluated under the same retained-view rate settings, evaluation metrics, and testing strategy.

Table 2 presents the clustering results for all methods across seven datasets. Based on the experimental results, we can draw the following conclusions. First, the proposed model demonstrates performance that is comparable to or significantly better than state-of-the-art baselines. Second, at low deletion rates, our method outperforms the baseline methods, and its performance remains stable even as deletion rates increase.

To explain these results, we analyze the key design aspects of our approach. Unlike autoencoder-based frameworks, our method employs VAEs as the overall architecture, which enhances the ability to learn probabilistic latent representations. This design improves robustness and generalization, particularly in handling incomplete data, contributing to the model's superior performance. Furthermore, the representation alignment strategy with dual constraints enables our model to learn more consistent and discriminative features compared to methods that rely solely on contrastive learning. This not only boosts clustering performance but also ensures stable performance across varying deletion rates. Finally, we note that the varying importance of different views plays a critical role in both data recovery and representation fusion. By dynamically adjusting view weights, our model optimizes the integration of multi-view information, further enhancing its ability to handle missing data and maintain performance stability.

Table 3
Ablation study on COIL20, 100Leaves, and handwritten datasets with a 30% paired/retained rate. Different methods use the modules marked with ✓ and ✗.

VAE	*FUSION*	*TRANSFER*	COIL20			100Leaves			handwritten		
			ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
✓	✗	✗	80.4	86.31	79.48	78.56	90.28	70.35	83	81.5	77.28
✗	✓	✗	80.33	87.45	80.6	77.5	89.05	65.16	90.1	82.82	81.29
✗	✗	✓	79.12	84.49	80.52	80.69	91.4	72.33	88.85	81.07	79.41
✓	✓	✗	81.26	89.17	81.33	78.94	90.03	68.39	91.3	82.92	81.33
✓	✗	✓	81.65	89.14	80.32	81.94	90.61	73.83	84	82.18	77.86
✗	✓	✓	82.08	90.49	81.74	81.18	90.97	71.48	91.7	84.46	81.98
✓	✓	✓	84.51	91.15	83.3	83.44	92.61	76.58	93.5	87.21	85.85

Table 4
Loss function ablation study on COIL20, 100Leaves, and handwritten datasets with a 30% paired/retained rate. Different methods use the modules marked with ✓ and ✗.

$\mathcal{L}_{VAE}$	$\mathcal{L}_{ALI}$	COIL20			100Leaves			handwritten		
		ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
✓	✗	80.69	86.31	75.99	70.13	84.04	57.33	78.75	80.15	76.3
✗	✓	82.1	88.91	82.35	44.25	66.8	27.81	53.27	60.64	58.37
✓	✓	84.51	91.15	83.3	83.44	92.61	76.58	93.5	87.21	85.85

5.3. Ablation study

In this subsection, a series of ablation experiments are performed to validate the effectiveness of the three key contributions of the present algorithm: the VAE encoding, the mutual information-guided data recovery, and the weighted representation fusion. First, the VAE encoding module is removed and replaced by the frequent use of AE for encoding. Next, the mutual information-guided data recovery module is removed in favor of missing instance completion based only on the relational graph without weighting. Finally, the weighted representation fusion module is removed in favor of the traditional additive and averaging approach to obtain the common representation. We perform ablation study experiments on COIL20, 100Leaves and handwritten datasets with a deletion rate = 0.3. Detailed results of the ablation experiments are shown in Table 3. Specifically, the columns “VAE”, “FUSION”, and “TRANSFER” in Table 3 indicate whether the variational autoencoder-based encoding module, the weighted representation fusion module, and the mutual information-guided transfer graph module are enabled, respectively. Three metrics, ACC, NMI and ARI, are used to evaluate this experiment.

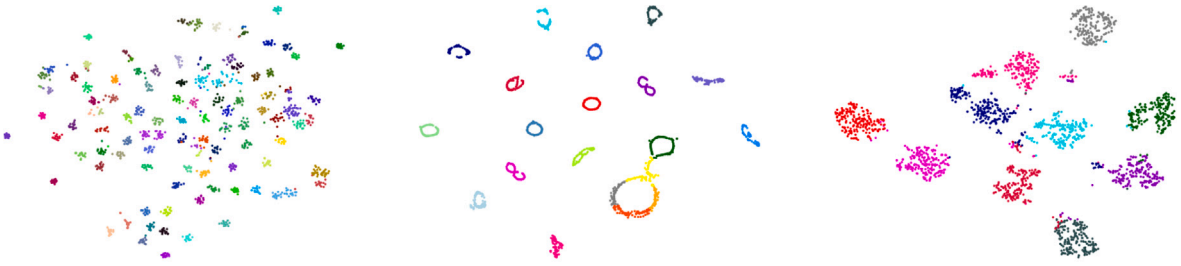
As can be seen from Table 4, the best performance is obtained only when all modules are used, indicating that all components are essential to the model. Moreover, when we fill in the missing instances using a transfer graph that is not weighted based on mutual information, we find that the effect is significantly lower than after weighting based on mutual information. Meanwhile, VAE plays an indispensable role in the whole model architecture, and from the results, we can see that when we replace VAE with AE for encoding, the effect decreases significantly. It is worth noting that the performance degradation caused by replacing VAE with AE is relatively less pronounced than that observed when removing the transfer or fusion modules. This is because both AE and VAE are capable of learning latent representations for individual views, whereas the primary advantage of VAE lies in its probabilistic latent modeling and regularization effect, which mainly improves representation smoothness, robustness, and generalization rather than directly maximizing clustering accuracy. However, adding the representation fusion module on top of this can help to fully utilize the consistency and complementarity information, as well as learn clustering-friendly representations. The ablation results demonstrate that each component is indispensable, and the full integration of these elements ensures that our model can exploit the latent information from each view, leading to more robust and accurate clustering outcomes.

To further analyze the effect of the alignment constraint, we conduct an additional ablation study at the loss-function level by selectively enabling the variational autoencoder loss $\mathcal{L}_{VAE}$ and the alignment loss $\mathcal{L}_{ALI}$. The results are reported in Table 4. When the alignment loss is removed and only $\mathcal{L}_{VAE}$ is used, the clustering performance consistently degrades across all datasets, indicating that reconstruction alone is insufficient to guarantee cross-view consistency.

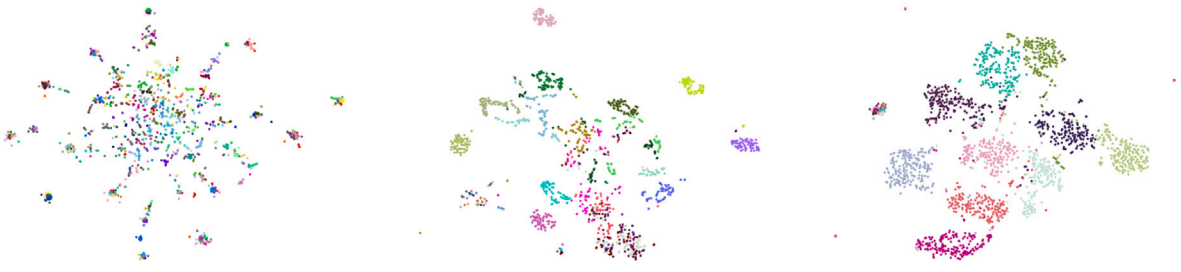
Moreover, when only $\mathcal{L}_{ALI}$ is optimized without the reconstruction constraint, the performance drops significantly, especially on the 100Leaves and handwritten datasets. This demonstrates that alignment without reliable latent reconstruction may lead to unstable or uninformative representations. In contrast, jointly optimizing $\mathcal{L}_{VAE}$ and $\mathcal{L}_{ALI}$ yields the best performance on all datasets, validating that the reconstruction and alignment losses are complementary and should be jointly considered to learn robust and discriminative multi-view representations.

5.4. Visualization analysis

To evaluate the effectiveness of our proposed method in comparison to other approaches, we utilize t-SNE to visualize the latent representations learned from different methods. We conduct this experiment on the COIL20, 100leaves and handwritten datasets with a missing rate of 0.3. As illustrated in Fig. 3, although the representations generated by existing methods like APADC show

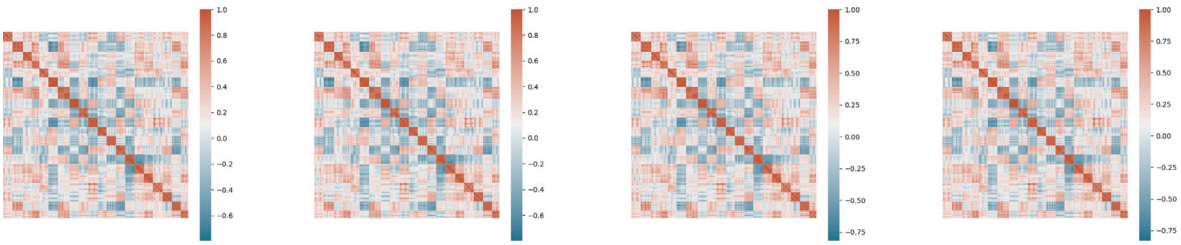


(a) ARLIMVC on 100leaves (b) ARLIMVC on COIL20 (c) ARLIMVC on handwritten



(d) APADC on 100leaves (e) APADC on COIL20 (f) APADC on handwritten

Fig. 3. T-SNE visualization of the learned representations of APADC and our method on the three datasets, respectively, where (a), (b), and (c) are our clustering results and (d), (e), and (f) belong to APADC with a pairwise view rate of 30%, where different colors represent different categories.



(a) Fused view (b) First view (c) Second view (d) Third view

Fig. 4. Visualization of the similarity matrix obtained from each view and fused view of COIL20 with a missing rate of 0.3.

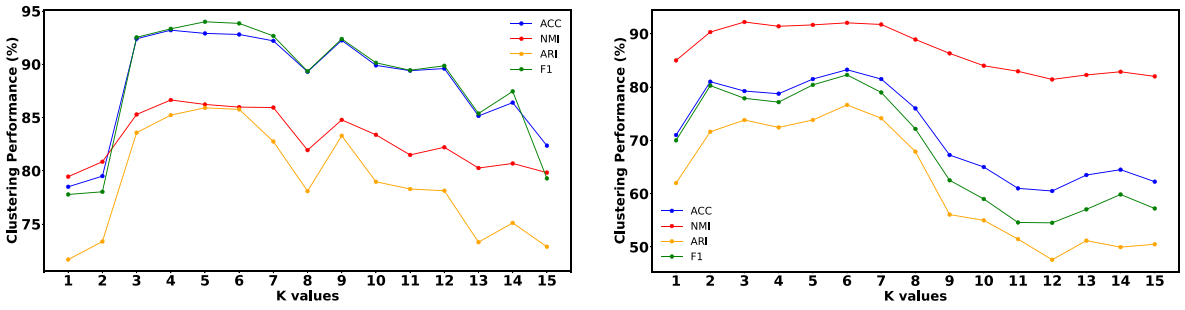
decent cluster separation, some overlap between clusters remains. In contrast, the representations from our method demonstrate more compact clusters with reduced intra-cluster variance and clearer separation between clusters compared to baseline models. This visualization underscores the ability of our model to learn robust and well-structured representations even when faced with incomplete multi-view data. This improvement can be attributed to our model's ability to fully leverage both consistent and complementary information across different views.

In addition, we computed similarity matrices from the latent representations using cosine distance, which are visualized in Fig. 4. The block-like structure of these matrices indicates high similarity among instances within the same cluster and lower similarity between clusters. The fused view's matrix presents a clearer and more well-defined structure compared to individual views, demonstrating the effectiveness of our fusion strategy. This result shows that our model's weighted representation fusion effectively enhances the complementarity between views, producing more discriminative representations.

5.5. Parameter sensitivity analysis

In our study, the main hyperparameter to consider during the graph construction stage for handling missing data is K, which determines the number of nearest neighbors. We perform a parameter sensitivity analysis for K on the handwritten and ORL datasets with a missing rate of 0.3, as shown in Fig. 5. We vary K from 1 to 15, as indicated on the x-axis, and evaluate clustering performance, depicted on the y-axis.

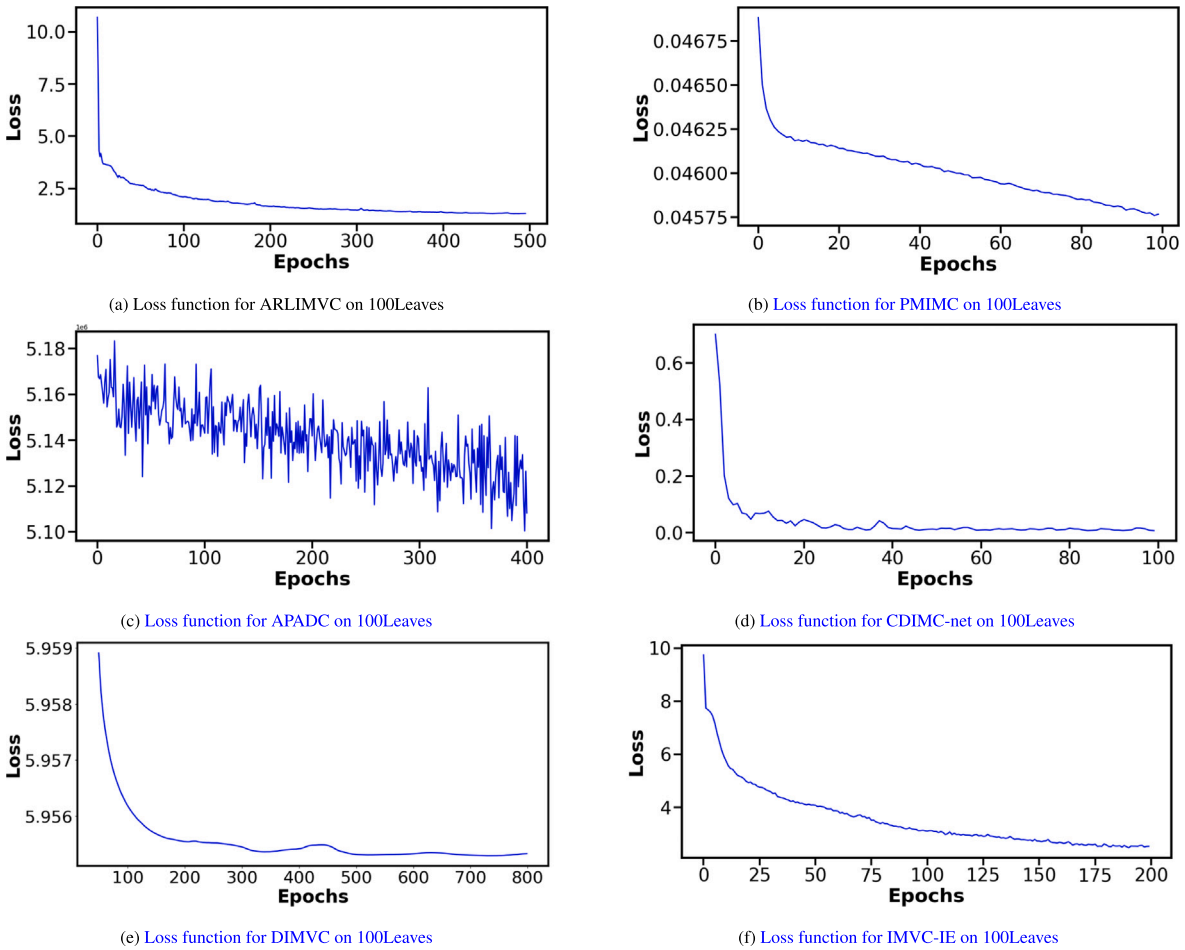
From Fig. 5, we observe that clustering performance degrades when K is set too small or too large. This behavior can be explained as follows: when K is too small, the graph fails to capture sufficient structural information due to the limited number of neighbors. Conversely, when K is too large, the graph incorporates irrelevant relationships, introducing noise and leading to incorrect adjacency relations.



(a) K sensitivity analysis on handwritten

(b) K sensitivity analysis on ORL

Fig. 5. The parameter analysis.



(a) Loss function for ARLIMVC on 100Leaves

(b) Loss function for PMIMC on 100Leaves

(c) Loss function for APADC on 100Leaves

(d) Loss function for CDIMC-net on 100Leaves

(e) Loss function for DIMVC on 100Leaves

(f) Loss function for IMVC-IE on 100Leaves

Fig. 6. Convergence analysis on 100Leaves dataset.

5.6. Convergence analysis

In Fig. 6, we show the loss function curve of the model with respect to the number of epochs on 100Leaves. In Fig. 6, we can see the loss value demonstrates a downward trajectory, with a notable decline occurring during the preliminary 200 epochs.

In addition, we conduct a comprehensive convergence analysis on the 100Leaves dataset by comparing the proposed ARLIMVC with several representative baseline methods, including PMIMC, APADC, CDIMC-net, DIMVC, and IMVC-IE. Fig. 6 presents the corresponding loss curves for all compared methods. As shown in the Fig. 6, most baseline methods (e.g., PMIMC and CDIMC-net) exhibit a rapid decrease in loss at the early training stage and converge within a relatively small number of epochs, while some methods such as APADC and DIMVC demonstrate slower or more fluctuating convergence behaviors. In contrast, ARLIMVC shows a relatively longer convergence process, with the loss decreasing steadily and smoothly over training epochs. Although ARLIMVC requires more

iterations to reach convergence, its optimization trajectory is stable and monotonic, without severe oscillations. This behavior can be attributed to the joint optimization of mutual information–guided data recovery, dual-constraint representation alignment, and adaptive fusion mechanisms. Overall, the convergence analysis highlights a trade-off between training efficiency and representation quality, and further confirms the stability and effectiveness of the proposed framework for incomplete multi-view clustering.

6. Conclusion and future work

In this paper, we propose a novel framework for incomplete multi-view clustering that effectively addresses the challenges of missing view recovery and makes full use of consistent and complementary information. To handle missing data, we introduce a mutual information-weighted relationship transfer mechanism combined with variational autoencoders, which not only completes the missing views but also enhances the quality of the recovered representations. Furthermore, we apply a distance-based cross-view local constraint to align representations across different views within the common representation space, and a contrastive learning based intra-view local constraint to improve the compactness of the distribution in this space. Additionally, we design a weighted representation fusion module based on Kolmogorov-Arnold networks to integrate multi-view data, leveraging complementary information.

Extensive experimental evaluations on benchmark datasets demonstrate that ARLIMVC achieves superior or highly competitive performance compared with state-of-the-art methods in most cases. These results validate the effectiveness of the proposed framework in capturing inter-view relationships, enforcing representation alignment, and integrating complementary information from incomplete multi-view data. In particular, ARLIMVC shows promising potential for practical applications in scenarios such as medical auxiliary diagnosis and multi-sensor industrial monitoring, where missing or incomplete observations are common.

Despite these advantages, we acknowledge that the current implementation of ARLIMVC incurs higher training time compared to some existing methods due to the inclusion of KAN-based fusion and dual-constraint alignment modules, which improve representation quality but increase optimization complexity. Improving computational efficiency and accelerating convergence, therefore constitute important directions for future research. Possible solutions include the design of more lightweight fusion architectures, more efficient optimization strategies, and parallelized or approximate training schemes. It is worth noting that the inference stage of ARLIMVC remains efficient and competitive once the model is trained.

Furthermore, the architectural design of ARLIMVC provides a flexible foundation for extension toward multi-task optimization (MTO) paradigms, where missing-view recovery, representation alignment, and clustering can be formulated as interrelated tasks within a unified framework. By leveraging advanced MTO strategies, such an extension may better balance generative and discriminative objectives, thereby enhancing robustness in highly incomplete scenarios and enabling broader applications such as joint cross-modal retrieval and classification.

CRediT authorship contribution statement

Yawen He: Writing – original draft, Software, Methodology, Conceptualization. **Mingjing Du:** Writing – review & editing, Writing – original draft, Funding acquisition.

Declaration of competing interest

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

Data availability

Data will be made available on request.

References

[1] X. Wan, X. Liu, J. Liu, S. Wang, Y. Wen, W. Liang, E. Zhu, Z. Liu, L. Zhou, Auto-weighted multi-view clustering for large-scale data, in: Proceedings of the 37th AAAI Conference on Artificial Intelligence, 2023, pp. 10078–10086.
[2] U. Fang, M. Li, J. Li, L. Gao, T. Jia, Y. Zhang, A comprehensive survey on multi-view clustering, IEEE Trans. Knowl. Data Eng. 35 (2023) 12350–12368.
[3] W. Jie, Y. Ke, Z. Zheng, X. Yong, Low-rank tensor graph learning based incomplete multi-view clustering, Acta Autom. Sin. 49 (2023) 1433–1445.
[4] Q. Wang, Z. Ding, Z. Tao, Q. Gao, Y. Fu, Generative partial multi-view clustering with adaptive fusion and cycle consistency, IEEE Trans. Image Process. 30 (2021) 1771–1783.
[5] Y. Wang, X. Yao, P. Zhu, W. Li, M. Cao, Q. Hu, Integrated heterogeneous graph attention network for incomplete multi-modal clustering, in: Proceedings of the 38th International Journal of Computer Vision, 2024, pp. 3847–3866.
[6] Y. Lin, Y. Gou, Z. Liu, B. Li, J. Lv, X. Peng, Completer: incomplete multi-view clustering via contrastive prediction, in: Proceedings of the 34th IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 11174–11183.
[7] W. Wu, G. Wen, L. Ou-Yang, R. Wang, S. Kwong, DUIMC: deep unbalanced incomplete multi-view clustering via graph constrained imputation and contrastive learning, in: Proceedings of the 33rd ACM International Conference on Multimedia, 2025, pp. 915–924.
[8] F. Ren, W. Chen, L. Gao, F. Guo, C. Liang, Dual-level distribution alignment for deep incomplete multi-view clustering, in: Proceedings of the 33rd ACM International Conference on Multimedia, 2025, pp. 2476–2485.
[9] Y. Zhang, L. Jiang, W. Liu, H. Yin, GloryIMVC: global-driven information theory for incomplete multi-view clustering, Knowl. Inf. Syst. 68 (2026) 63.
[10] H. Yuan, Y. Sun, F. Zhou, J. Wen, S. Yuan, X. You, Z. Ren, Prototype matching learning for incomplete multi-view clustering, IEEE Trans. Image Process. 34 (2025) 828–841.
[11] Y. Zhang, Y. Lin, W. Yan, L. Yao, X. Wan, G. Li, C. Zhang, G. Ke, J. Xu, Incomplete multi-view clustering via diffusion contrastive generation, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2025, pp. 22650–22658.

[12] X. Li, G. Li, X. Zhang, Y. Wang, Q. Shi, W. Liang, Incomplete multi-view clustering via local reasoning and correlation analysis, in: Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining, 2025, pp. 484–492.
[13] B. Jiang, C. Zhang, X. Liang, P. Zhou, J. Yang, X. Wu, J. Guan, W. Ding, W. Sheng, Collaborative similarity fusion and consistency recovery for incomplete multi-view clustering, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2025, pp. 17617–17625.
[14] Q. Wang, M. Du, Structure-aware granular ball clustering, Inf. Sci. 740 (2026) 123224.
[15] M. Yin, R. Chen, R. Lin, Y. Wang, Multi-ssalvcae: self-supervised adversarial learning-based view-common latent autoencoders for multiview clustering, IEEE Trans. Syst. Man Cybern. Syst. 54 (2024) 5456–5467.
[16] J. Xu, Y. Ren, H. Tang, Z. Yang, L. Pan, Y. Yang, X. Pu, P.S. Yu, L. He, Self-supervised discriminative feature learning for deep multi-view clustering, IEEE Trans. Knowl. Data Eng. 35 (2023) 7470–7482.
[17] J. Jin, S. Wang, Z. Dong, X. Liu, E. Zhu, Deep incomplete multi-view clustering with cross-view partial sample and prototype alignment, in: Proceedings of the 36th IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 11600–11609.
[18] S.-Y. Li, Y. Jiang, Z.-H. Zhou, Partial multi-view clustering, in: Proceedings of the 28th AAAI Conference on Artificial Intelligence, 2014, pp. 1968–1974.
[19] J. Wen, Z. Zhang, Y. Xu, Z. Zhong, Incomplete multi-view clustering via graph regularized matrix factorization, in: European Conference on Computer Vision, 2018, pp. 593–608.
[20] X. Liu, M. Li, C. Tang, J. Xia, J. Xiong, L. Liu, M. Kloft, E. Zhu, Efficient and effective regularized incomplete multi-view clustering, IEEE Trans. Pattern Anal. Mach. Intell. 43 (2021) 2634–2646.
[21] J. Guo, J. Ye, Anchors bring ease: an embarrassingly simple approach to partial multi-view clustering, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2019, pp. 118–125.
[22] A. Trivedi, P. Rai, H. Daumé III, S.L. DuVall, Multiview clustering with incomplete views, in: Proceedings of the 24th Conference on Neural Information Processing Systems, 2010, pp. 1–8.
[23] J. Wen, K. Yan, Z. Zhang, Y. Xu, J. Wang, L. Fei, B. Zhang, Adaptive graph completion based incomplete multi-view clustering, IEEE Trans. Multimed. 23 (2020) 2493–2504.
[24] W. Hao, S. Pang, X. Bai, J. Xue, Tensor-based incomplete multi-view clustering with low-rank data reconstruction and consistency guidance, IEEE Trans. Circuits Syst. Video Technol. 33 (2023) 7156–7169.
[25] X. Liu, S. Ma, G. Chen, J. Liu, L. Zong, Cross-view alignment and completion for incomplete information multi-view clustering, Inf. Sci. 720 (2025) 122515.
[26] M. Li, R. Zhang, Y. Zhang, X. Piao, S. Zhao, B. Yin, SCAE: structural contrastive auto-encoder for incomplete multi-view representation learning, ACM Transactions on Multimedia Computing, Communications and Applications 20 (2024) 1–24.
[27] K. Zhang, Y. Wang, S. Gao, S. Du, X. He, Direct contrastive learning for incomplete multi-view clustering, in: Proceedings of the 36th Chinese Control and Decision Conference, 2024, pp. 6228–6233.
[28] A. Li, C. Feng, S. Xu, Y. Cheng, Graph t-SNE multi-view autoencoder for joint clustering and completion of incomplete multi-view data, Knowl.-based Syst. 284 (2024) 111324.
[29] J. Pu, C. Cui, X. Chen, Y. Ren, X. Pu, Z. Hao, S.Y. Philip, L. He, Adaptive feature imputation with latent graph for deep incomplete multi-view clustering, in: Proceedings of the 38th AAAI Conference on Artificial Intelligence, 2024, pp. 14633–14641.
[30] Y. Wang, D. Chang, Z. Fu, J. Wen, Y. Zhao, Graph contrastive partial multi-view clustering, IEEE Trans. Multimed. 25 (2022) 6551–6562.
[31] J. Xu, C. Li, Y. Ren, L. Peng, Y. Mo, X. Shi, X. Zhu, Deep incomplete multi-view clustering via mining cluster complementarity, in: Proceedings of the 36th AAAI Conference on Artificial Intelligence, 2022, pp. 8761–8769.
[32] Z. Liu, Y. Wang, S. Vaidya, F. Ruehle, J. Halverson, M. Soljačić, T.Y. Hou, M. Tegmark, KAN: Kolmogorov-Arnold networks, arxiv:2404.19756, 2024.
[33] H. Zhao, H. Liu, Y. Fu, Incomplete multi-modal visual data grouping, in: Proceedings of the 25th International Joint Conference on Artificial Intelligence, 2016, pp. 2392–2398.
[34] J. Wen, Z. Zhang, Y. Xu, B. Zhang, L. Fei, G.-S. Xie, CDIMC-net: cognitive deep incomplete multi-view clustering network, in: Proceedings of the 29th International Conference on International Joint Conferences on Artificial Intelligence, 2021, pp. 3230–3236.
[35] B. Huang, Z. Huang, S. Lan, Q. Zheng, Y. Yu, Incomplete multi-view clustering via inference and evaluation, in: Proceedings of the 49th IEEE International Conference on Acoustics, Speech and Signal Processing, 2024, pp. 8180–8184.
[36] M. Hu, S. Chen, Doubly aligned incomplete multi-view clustering, in: Proceedings of the 27th International Joint Conference on Artificial Intelligence, 2019, pp. 2262–2268.
[37] J. Xu, C. Li, L. Peng, Y. Ren, X. Shi, H.T. Shen, X. Zhu, Adaptive feature projection with distribution alignment for deep incomplete multi-view clustering, IEEE Trans. Image Process. 32 (2023) 1354–1366.