Dual Contrast-Driven Deep Multi-View Clustering

Jinrong Cui, Yuting Li, Han Huang, *Senior Member, IEEE*, and Jie Wen, *Senior Member, IEEE*

***Abstract*— Consensus representation learning is one of the most popular approaches in the field of multi-view clustering. However, most of the existing methods cannot learn discriminative representations with a clustering-friendly structure since these methods ignore the separation among clusters and the compactness within each cluster. To tackle this issue, we propose a new deep multi-view clustering network with a dual contrastive mechanism to learn clustering-friendly representations. Specifically, our method employs dual contrasting losses: a dynamic cluster diffusion loss to maximize the distance between different clusters and a reliable neighbor-guided positive alignment loss to enhance compactness within each cluster. Our approach includes several key components: view-specific encoders to extract high-level features from each view, and an adaptive feature fusion strategy to obtain consensus representations across multiple views. The dynamic cluster diffusion module ensures inter-cluster separation by maximizing distances between different clusters in the consensus feature space. Simultaneously, the reliable neighbor-guided positive alignment module improves within-cluster compactness through a pseudo-label and nearest neighbor structure-driven contrastive loss. Experimental results on several datasets show that our method can acquire clustering-friendly representations with both good properties of inter-cluster separation and within-cluster compactness, and outperforms the existing state-of-the-art approaches in clustering performance. Our source code is available at https://github.com/tweety1028/DCMVC.**

***Index Terms*— Multi-view clustering, deep clustering, representation learning, contrastive learning.**

I. Introduction

CLUSTERING is a classic task in unsupervised learning and serves the purpose of categorizing samples into different clusters without the aid of label information. Clustering plays a crucial role in various real-world applications such as data mining [1], [2], [3], [4], image segmentation [5], [6], and machine learning [7], [8], [9], [10], [11]. With advancements in data acquisition and storage, massive data collected in real-world scenarios often involve information gathered from various perspectives or sensor types, commonly termed multi-view data or multi-modal data [12]. Labeling an extensive volume of multi-view data is an exceedingly time-consuming and labor-intensive undertaking. Consequently, multi-view clustering (MVC) has emerged as a hot topic in research and applications.

Traditional MVC methods include kernel-based [13], [14], [15], subspace learning [16], [17], [18], [19], and graph-based approaches [20], [21], [22]. Kernel-based methods employ kernel techniques to handle the data with nonlinear relationships, which commonly map samples into a higher-dimensional space to facilitate linear clustering operations in that space. Subspace learning methods generally transform data into a low-dimensional subspace and try to capture shared subspace information across various views. Graph-based methods seek to learn a consensus graph or several view-specific graphs that reflect the intrinsic similarity relationships among samples and then calculate the clustering results according to the graph partition theory. Although traditional MVC methods have shown impressive effectiveness and generally have meaningful learning models, the poor feature extraction ability limits their performance. In addition, many traditional MVC methods, especially graph-based methods, generally have high computational complexities, thereby restricting their applicability in certain scenarios.

Given the powerful capabilities of deep learning in nonlinear transformations and high-dimensional data processing, researchers have proposed a succession of deep MVC methods [23], [24], [25], [26], [27]. The existing deep MVC methods can be simply categorized into two types: non-contrastive [28], [29], [30] and contrastive-based [24], [31], [32] approaches. Non-contrastive methods tend to extract shared information from different views, fostering stronger consistency in sample representations through holistic learning. Contrastive-based methods, by introducing contrastive loss or frameworks, encourage samples of positive pairs to be closer in the embedding space and push samples of negative pairs farther away. This emphasizes the discriminative degree of the learned representation in the embedding space.

Although notable advancements have been made by current MVC methods over the past decade, their performance is still limited owing to the following issues: Some methods excessively emphasize shared information from different views. They overlook the necessity of enlarging the differences between samples in different clusters and bringing samples in the same cluster closer together. Because of these issues, these methods cannot achieve optimal discriminative representations with a clustering-friendly structure. Although existing contrastive-based methods generally outperform non-contrastive methods, they still have limitations. Specifically, they often overlook the importance of maintaining inter-cluster

Manuscript received 24 January 2024; revised 15 July 2024; accepted 27 July 2024. Date of publication 26 August 2024; date of current version 30 August 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62372136 and in part by Guangdong Basic and Applied Basic Research Foundation under Grant 2024A1515030213. The associate editor coordinating the review of this article and approving it for publication was Dr. Junhui Hou. *(Corresponding authors: Han Huang; Jie Wen.)*

Jinrong Cui and Yuting Li are with the College of Mathematics and Informatics, South China Agricultural University, Guangzhou 510620, China (e-mail: tweety1028@163.com; liyuting3512@gmail.com).

Han Huang is with the School of Software Engineering, South China University of Technology, Guangzhou 510642, China (e-mail: hhan@scut.edu.cn).

Jie Wen is with Shenzhen Key Laboratory of Visual Object Detection and Recognition, Harbin Institute of Technology, Shenzhen, Shenzhen, 518055, China (e-mail: jiewen_pr@126.com).

This article has supplementary downloadable material available at https://doi.org/10.1109/TIP.2024.3444269, provided by the authors.

Digital Object Identifier 10.1109/TIP.2024.3444269

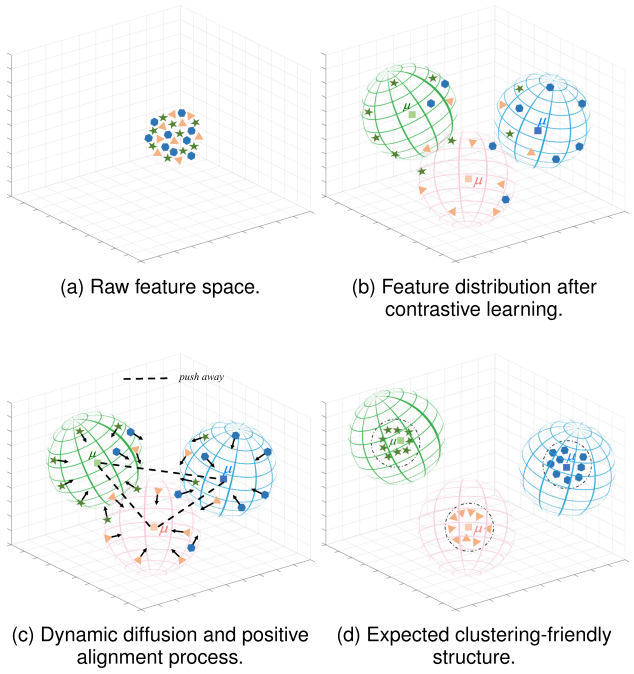


Fig. 1. A representative example showcasing our motivation.

separation and within-cluster compactness. Moreover, these methods generally introduce false-negative pairs, which negatively impact clustering performance. For example, they typically consider representations of different views of the same sample as positive pairs, while treating all other representations as negative pairs, which is overly arbitrary since this treatment ignores the within-cluster relationship.

To address the aforementioned issues, we propose a new Dual Contrastive learning-based deep Multi-View Clustering network (DCMVC) in this paper. Our motivation is derived from the insights illustrated in Fig. 1. As illustrated in Fig. 1a, data points in raw feature space are difficult to categorize. By introducing contrastive learning, the distribution of data points will exhibit clear discriminative characteristics, as shown in Fig. 1b. However, different clusters cannot be completely separated and are still coupled with each other to some extent. Simultaneously, samples in the same cluster are dispersed, indicating relatively low compactness. To solve this issue, in our work, as shown in Fig. 1c, we try to design a new contrastive learning module that can simultaneously push different clusters away and pull samples in the same cluster together, resulting in well-separated inter-cluster and compacted within-cluster structure as shown in Fig. 1d.

Based on the above motivation, we design the DCMVC network by integrating four major modules as shown in Fig. 2: view-specific autoencoders, adaptive feature fusion module, dynamic cluster diffusion module, and reliable neighbor-guided positive alignment module. View-specific autoencoders extract sufficient high-level features from each view. The adaptive feature fusion module is introduced to harness the complementary information from multiple views and produce the consensus representation to obtain a unique clustering result for data. To push different clusters far away, a dynamic cluster diffusion module is added, which treats different clusters as negative pairs and introduces a cluster center-based contrastive loss. To learn more reliable and compacted representations in each cluster, we further introduce the reliable neighbor-guided positive alignment module, which aims at eliminating the negative influence caused by false-negative pairs and learning a high discriminative representation with the clustering-friendly structure to enhance the performance. Compared with the existing works, our work has the following contributions:

- We propose an end-to-end deep multi-view clustering method, termed DCMVC, which introduces the dual contrastive mechanism to learn the discriminative representation with clustering-friendly structure like well-separated clusters and compacted within-cluster.
- We propose a dynamic cluster diffusion module, which introduces a new cluster-level contrastive loss to align the clusters' representation of all views and enlarge the inter-cluster distribution, thereby forming well-separated clusters.
- We propose a new reliable neighbor-guided positive alignment module and design an instance-level contrastive loss. It enables the network to obtain a more discriminative representation with compacted within-cluster structure and separated inter-cluster structure by sufficiently considering the intrinsic nearest neighbor structure information to guide the network training and eliminate the negative influence of false-negative pairs.

II. Related Work

In the realm of representation learning-based MVC family [33], the existing methods predominantly fall into two categories: shallow representation learning-based method and deep representation learning-based method. This section briefly reviews some representative works of shallow representation learning-based methods and deep representation learning-based methods. In addition, we also introduce contrastive learning, which is widely exploited in many unsupervised learning works.

A. Contrastive Learning

Contrastive learning is an unsupervised learning method, which performs instance-wise discrimination using the normalized cross-entropy as information (InfoNCE) loss [34], [35], [36]. It begins by constructing positive and negative pairs for each instance, followed by maximizing the similarities among positive pairs and minimizing those among the negatives in a latent feature subspace. In the construction of pairs, researchers have developed various methods. For instance, [37] proposes to augment the data and construct positive and negative pairs in the mini-batches. The augmented instances of the same sample constitute positive pairs, while all other pairings form negative pairs. So far, researchers have put forth various contrastive learning methods that work effectively without the requirement of negative pairs. For example, in [38], the loss on negative pairs is replaced by maximizing the similarity between the predictions of an autoencoder and a target network.

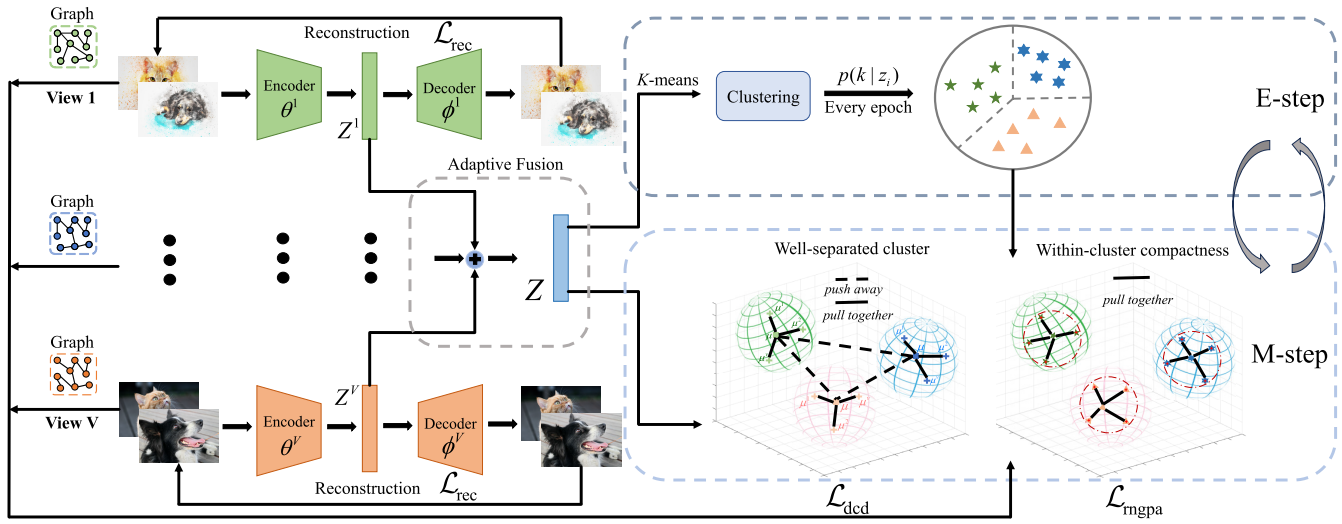


Fig. 2. The overall framework of the proposed DCMVC within an Expectation-Maximization framework. The framework includes: (a) View-specific Autoencoders and Adaptive Feature Fusion Module, which extracts high-level features and fuses them into consensus representations (with loss $\mathcal{L}_{\text{rec}}$); (b) Dynamic Cluster Diffusion Module, enhancing inter-cluster separation by maximizing the distance between clusters (with loss $\mathcal{L}_{\text{dcd}}$); (c) Reliable Neighbor-guided Positive Alignment Module, improving within-cluster compactness using a pseudo-label and nearest neighbor structure-driven contrastive learning (with loss $\mathcal{L}_{\text{rngpa}}$); (d) Clustering-friendly Structure, ensuring well-separated and compact clusters.

Owing to the remarkable success of contrastive learning in the unsupervised domain, researchers have introduced it to the clustering field and proposed a variety of contrastive learning based deep clustering methods [39], [40], [41]. For example, [39] establishes positive and negative pairs through data augmentation. Simultaneously, it optimizes the instance- and cluster-level contrastive loss to maximize the similarity of positive pairs and minimize the similarity of negative pairs. Reference [40] introduces prototype scattering loss and positive sampling alignment module to address the class collision issue. However, these methods are all single-view clustering methods and cannot handle multi-view clustering tasks. Reference [41] presents a superpixel graph contrastive clustering model for hyperspectral image clustering. By leveraging contrastive learning with semantic-invariant augmentations, it enhances superpixel representation and significantly improves clustering accuracy.

B. Shallow Representation Learning-Based Method

The shallow learning-based MVC methods can be further classified into two main groups: multi-view subspace clustering and multi-view graph clustering [33]. Generally speaking, multi-view subspace clustering methods learn a unified subspace representation from specific subspaces of all views. For example, [42] designs an innovative angular-based regularization term to achieve multiple views data association consensus. Reference [19] introduces an innovative tensor low-rank representation method specifically designed for spectral clustering in multi-view settings. This approach effectively captures and integrates inter- and intra-view relationships within a unified framework. Different from multi-view subspace clustering, multi-view graph clustering methods try to construct view-specific graphs for each view and then obtain a shared graph through regularization terms. For instance, [21] learns individual view graphs through an iterative cross-diffusion process and derives the final unified clustering graph by averaging these refined view-related graphs. Reference [43] helps the learning of each view graph matrix and the coherent graph matrix in a mutually reinforcing manner, and automatically weights each view graph matrix to produce the coherent graph matrix.

Although shallow representation learning-based MVC methods have demonstrated promising results, they are difficult to capture complex hierarchical representations present in high-dimensional data. In addition, owing to the limitations of shallow methods on feature extraction, these methods may be ineffective in dealing with the data with nonlinear relationships.

C. Deep Representation Learning-Based Method

Considering the significant advancements of deep learning in unsupervised domains, many researchers devoted themselves to deep representation learning-based MVC. These studies can be simply grouped into two categories: non-contrastive and contrastive-based approaches. Non-contrastive methods typically directly optimize objective loss related to clustering tasks during training, aiming to explore consistency and complementary information of the multi-view data [44], [45]. For instance, driven by the observation that multi-view data possesses a shared latent embedding, [28] proposes the learning of a generative latent representation that adheres to a mixture of Gaussian distributions. Reference [29] disentangles the view-common and view-peculiar representations by controlling the mutual information to mine common discrete cluster information. Reference [46] enforces a diagonal constraint on the consensus representation obtained through multiple autoencoders with a self-expression learning scheme. Although non-contrastive methods have obtained notable achievements, they generally cannot obtain sufficiently discriminative representations with clustering-friendly structure.

TABLE I
DETAILS INFORMATION OF NOTATIONS

Notations	Descriptions
$\mathbf{X}$	The multi-view dataset
N	Number of samples
V	Number of views
M	Batch size
d_v	Dimension of input data from view v
$X^v \in \mathbb{R}^{N\times d_v}$	Input data from view v
$\boldsymbol{x}_i^v \in \mathbb{R}^{d_v}$	Vector feature of the i-th sample in the v-th view
$\hat{\boldsymbol{x}}_i^v \in \mathbb{R}^{d_v}$	Reconstruction feature of the i-th sample in the v-th view
$\boldsymbol{z}_i^v \in \mathbb{R}^d$	Extracted representation of $\boldsymbol{x}_i^v$, with d being the dimension
$\boldsymbol{z}_i \in \mathbb{R}^d$	The consensus representation of the i-th sample.
$\boldsymbol{a}_v \in \mathbb{R}^1$	Trainable weight parameter for the v-th view
$\boldsymbol{w}_v \in \mathbb{R}^1$	Normalized weight parameter for the v-th view
$\boldsymbol{\mu}_k^v \in \mathbb{R}^d$	Clusters' representations of k-th cluster in v-th view feature space
$\boldsymbol{\mu}_k \in \mathbb{R}^d$	Clusters' representations of k-th cluster in the consensus feature space
$\mathcal{G}^v$	The nearest-neighbor graph constructed from v-th view
$\boldsymbol{Y} \in \mathbb{R}^{N\times N}$	Indicator matrix denoting if two samples are assigned to the same cluster
$\mathcal{P}_i^v$	Positive pair set of i-th instance in the v-th view
$\mathcal{N}_i^v$	Negative pair set of i-th instance in the v-th view

In recent years, as a hot topic in unsupervised learning, contrastive learning is also introduced to the deep multi-view clustering fields. For example, [47] proposes a view-wise contrastive learning method to address the challenging issue of the learned representation with only fewer separable clusters. It treats views of the same sample as positive pairs and views of different samples as negative pairs. Reference [32] introduces feature-level alignment-oriented, commonality-oriented, and cluster-level consistency-oriented contrastive learning modules to compare representations at different feature-level and cluster-level. Reference [48] improves cluster assignment accuracy by constraining the clustering allocations across multiple views. The method can capture consistent semantic label information of multiple views. These contrastive-based MVC methods generally obtain better performance than the previous methods without contrastive learning. However, the existing contrastive learning-based methods still suffer from the issue of false-negative pairs, resulting in unreliable clustering performance.

III. THE PROPOSED METHOD

Task Statement: Let $\mathbf{X} = \{X^v \in \mathbb{R}^{N\times d_v}\}_{v=1}^V$ be a multi-view dataset, where V stands for the total number of views, N denotes the number of samples. $d_1, d_2, \ldots,$ and d_V denote the feature dimensions of the corresponding views. The target of MVC is to precisely group these N samples into K disjoint clusters. The Table I provides detailed information about the variables used in our model, including their descriptions and dimensions.

For the above MVC task, we propose a new deep MVC network, called DCMVC. As stated in the introduction and shown in Fig. 2, the proposed network is composed of four major modules. In this section, we will present the four modules, overall objective loss, and training strategy in detail.

A. View-Specific Autoencoders and Adaptive Feature Fusion Module

For unsupervised MVC tasks, it is crucial to extract the discriminative features from the original multi-view data with diverse feature dimensions and learn the consensus representation shared by all views to obtain a unique clustering result. In view of the good property of deep neural networks like autoencoder in unsupervised feature extraction and considering the diverse view types with different feature dimensions, we introduce several view-specific autoencoders, in which the encoder modules can extract the deep-level features of all views and the decoder modules enable the extracted view-specific features to preserve more information of their original data of all views, respectively. Specifically, let f^v and g^v be the encoder and decoder for the v-th view. $\boldsymbol{\theta}^v$ and $\boldsymbol{\phi}^v$ represent the parameters of the encoder and the decoder of the v-th view. The latent informative representation of the i-th sample extracted by the encoder of the v-th view can be formulated as:

$$\boldsymbol{z}_i^v = f^v(\boldsymbol{x}_i^v, \boldsymbol{\theta}^v), \tag{1}$$

where $\boldsymbol{z}_i^v \in \mathbb{R}^d$ is the extracted representation of $\boldsymbol{x}_i^v$. d denotes the dimension of the latent representation. Taking the v-th view as an example, the decoder process can be formulated as:

$$\hat{\boldsymbol{x}}_i^v = g^v(\boldsymbol{z}_i^v, \boldsymbol{\phi}^v) = g^v(f^v(\boldsymbol{x}_i^v, \boldsymbol{\theta}^v), \boldsymbol{\phi}^v), \tag{2}$$

where $\hat{\boldsymbol{x}}_i^v$ denotes the reconstructed data of the v-th view decoder for the i-th sample.

Similar to the conventional autoencoder, the reconstruction loss is introduced for all views to compel their autoencoders to capture the informative deep-level features and reduce the information loss. The reconstruction loss of all views can be formulated as follows:

$$\mathcal{L}_{\text{rec}} = \sum_{v=1}^{V}\sum_{i=1}^{N}\left\|\boldsymbol{x}_i^v - \hat{\boldsymbol{x}}_i^v\right\|_2^2. \tag{3}$$

For an input sample with multiple views, these view-specific encoders will generate several view-specific latent representations with many complementary information. To obtain a consensus and good clustering result, two popular techniques will be adopted to obtain a consensus latent representation: concatenation and fusion. Inspired by the motivation that the fusion approach can simultaneously consider consistency and complementarity of multi-view data [49], we choose the fusion approach to obtain the consensus latent representation. Specifically, considering that different views may contain varying amounts of information, we introduce an adaptive representation fusion strategy as follows:

$$\boldsymbol{z}_i = \sum_{v=1}^{V}\boldsymbol{w}_v\boldsymbol{z}_i^v = \sum_{v=1}^{V}\frac{e^{\boldsymbol{a}_v}}{\sum_{l=1}^{V}e^{\boldsymbol{a}_l}}\boldsymbol{z}_i^v. \tag{4}$$

where $\boldsymbol{z}_i$ denotes the consensus representation of the i-th sample. The weight $\boldsymbol{w}_v$ is adaptively determined by the trainable parameter $\boldsymbol{a}_v$ for each view. These parameters are used in a softmax function to calculate the normalized weights: $\boldsymbol{w}_v = \sum_{v=1}^{V}\frac{e^{\boldsymbol{a}_v}}{\sum_{l=1}^{V}e^{\boldsymbol{a}_l}}$. $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_V$ can be regarded as the normalized

weights to these view-specific representations, which satisfy $\boldsymbol{w}_v > 0$ and $\sum_{v=1}^{V} \boldsymbol{w}_v = 1$.

The introduction of these learnable weights enables a comprehensive and adaptive adjustment of each view's influence on the consensus feature space and the clustering result. This guarantees that the model can dynamically explore the distinct characteristics of multi-view data, providing a more comprehensive and adaptive representation.

B. Dynamic Cluster Diffusion Module

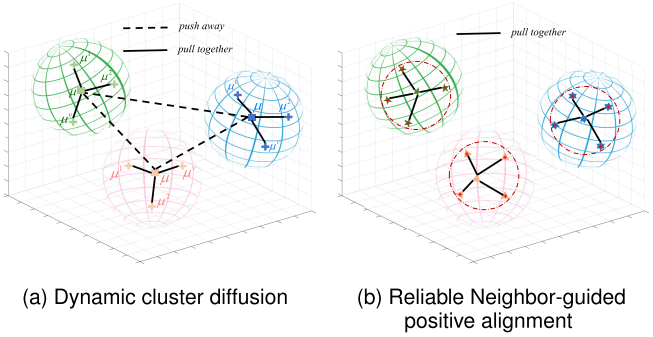
For unsupervised clustering tasks, it is expected to obtain distinct and well-separated clusters. To this end, inspired by the remarkable success of contrastive learning [40], we introduce a new dynamic cluster diffusion module (DCD), which seeks to simultaneously emphasize cluster cohesion across multiple views and promote inter-cluster separation in the latent representation space. Specifically, assuming that the multi-view dataset consists of K categories/clusters or the data is expected to be grouped into K clusters, $\boldsymbol{\mu}_i$ and $\boldsymbol{\mu}_i^v$ denote the representation of the i-th cluster in the consensus representation space and the v-th view-specific representation space, respectively. We design the following cluster-driven contrastive loss, inspired by the decoupled contrastive learning approach outlined in [50]:

$$\begin{aligned}\mathcal{L}_{\text{dcd}} = &\frac{1}{K}\sum_{v=1}^{V}\sum_{k=1}^{K} \\ &-\log\frac{\exp\left(\frac{s(\boldsymbol{\mu}_k,\boldsymbol{\mu}_k^v)}{\tau_C}\right)}{\exp\left(\frac{s(\boldsymbol{\mu}_k,\boldsymbol{\mu}_k^v)}{\tau_C}\right)+\sum_{\substack{j=1\\ j\neq k}}^{K}\exp\left(\frac{s(\boldsymbol{\mu}_k,\boldsymbol{\mu}_j)}{\tau_C}\right)} \\ \approx &\underbrace{\frac{1}{K}\sum_{v=1}^{V}\sum_{k=1}^{K}-\frac{s\left(\boldsymbol{\mu}_k,\boldsymbol{\mu}_k^v\right)}{\tau_C}}_{\text{cluster cohesion}} \\ &+\underbrace{\frac{1}{K}\sum_{v=1}^{V}\sum_{k=1}^{K}\log\sum_{\substack{j=1\\ j\neq k}}^{K}\exp\left(\frac{s\left(\boldsymbol{\mu}_k,\boldsymbol{\mu}_j\right)}{\tau_C}\right)}_{\text{cluster separation}},\end{aligned} \tag{5}$$

where τ_C serves as the temperature parameter. $s(\cdot,\cdot)$ denotes the similarity function, which is defined as $s(\boldsymbol{\mu}_k,\boldsymbol{\mu}_j) = \frac{\boldsymbol{\mu}_k^{\mathrm{T}}\boldsymbol{\mu}_j}{\|\boldsymbol{\mu}_k\|\|\boldsymbol{\mu}_j\|}$. The similarity function measures the cosine similarity between clusters' representations. The temperature parameter controls the sharpness of the distribution. The loss function $\mathcal{L}_{\text{dcd}}$ enforces cluster cohesion (pulls positive pairs closer) and separation (pushes negative pairs apart). In our method, for a mini-batch $\mathcal{B}$ of the multi-view data, clusters' representations $\boldsymbol{\mu}_k$ and $\boldsymbol{\mu}_k^v$ are updated as follows:

$$\boldsymbol{\mu}_k = \frac{\sum_{z_i\in\mathcal{B}} p(k|z_i)z_i}{\left\|\sum_{z_i\in\mathcal{B}} p(k|z_i)z_i\right\|_2}, \tag{6}$$

$$\boldsymbol{\mu}_k^v = \frac{\sum_{z_i^v\in\mathcal{B}} p(k|z_i)z_i^v}{\left\|\sum_{z_i^v\in\mathcal{B}} p(k|z_i)z_i^v\right\|_2}. \tag{7}$$



(a) Dynamic cluster diffusion (b) Reliable Neighbor-guided positive alignment

Fig. 3. Illustration of the proposed key techniques in DCMVC.

where $p(k|z_i)$ represents the hard assignment of the i-th sample belonging to the k-th cluster. During training, obtaining accurate $p(k|z_i)$ is crucial for optimizing the proposed model. Hence, we adopt an Expectation-Maximization (EM) framework that alternately utilizes K-means clustering at every epoch in the E-step. Subsequently, in the M-step, we minimize the objective loss to optimize the model. This process will be detailed later.

Concretely, as shown in Eq. (5), the cluster-driven contrastive loss can be roughly decomposed into two major components: *cluster cohesion* and *cluster separation*. As shown in Fig. 3a, on one hand, the cluster cohesion aims to align the clusters across multiple views, which can promote the consistency across multiple views. On the other hand, minimizing the cluster separation term encourages different clusters to be pushed away in the latent feature space, which yields distinct and well-separated cluster structure.

C. Reliable Neighbor-Guided Positive Alignment Module

As previously highlighted, negative pairs play a pivotal role in contrastive-based MVC methods, facilitating the acquisition of discriminative representations. However, the conventional contrastive learning approaches only treat multiple views of the same sample as positive pairs while regarding all of the other views as negative pairs no matter whether these views belong to the same class. This is obviously unreasonable because it goes against the expectation of within-cluster compactness. To solve this issue and obtain a compacted structure for the within-cluster data as shown in Fig. 3b, we propose a reliable neighbor-guided positive alignment (RNGPA) module. Different from the existing contrastive learning methods, RNGPA constructs the positive and negative pairs by sufficiently taking into account the nearest neighbor and pseudo-clustering-label information.

Intuitively, samples that are close to each other in the original feature space are more likely to belong to the same cluster. This neighbor information has been validated to be useful in enhancing the clustering performance [24]. Therefore, we will introduce the neighbor information of all views to guide the model training. For the v-th view, the nearest-neighbor graph $\mathcal{G}^v$ is constructed as follows:

$$\mathcal{G}_{i,j}^v = \begin{cases} 1 & (\boldsymbol{x}_i^v \in \varphi(\boldsymbol{x}_j^v)) \ or \ (\boldsymbol{x}_j^v \in \varphi(\boldsymbol{x}_i^v)) \\ 0 & otherwise, \end{cases} \tag{8}$$

where $\varphi(\boldsymbol{x}_i^v)$ denotes the nearest instance set to $\boldsymbol{x}_i^v$.

After obtaining the nearest neighbors for each sample, a straightforward way is to consider all neighbors of a sample as positive pairs. However, such an intuitive approach may introduce noisy information because some selected nearest neighbors may not genuinely belong to the same cluster as the reference sample. To avoid the negative influence of incorrect neighbor pair information, we rely on the hard assignment of clusters obtained in the previous subsection to ensure the reliability of those positive pairs. Specifically, in our model, a positive pair is constructed only when two samples are nearest neighbors and are assigned to the same cluster. Conversely, a negative pair is constructed when two samples are neither nearest neighbors nor come from the same cluster.

According to the above analysis, we first perform K-means on the consensus representation $\boldsymbol{Z}$ to obtain an indicator matrix $\boldsymbol{Y} \in \mathbb{R}^{N\times N}$ to indicate whether two samples are assigned to the same cluster. If the i-th and j-th samples are assigned to the same cluster, $\boldsymbol{Y}_{i,j} = 1$; otherwise, $\boldsymbol{Y}_{i,j} = 0$. Then, taking a mini-batch data $\mathcal{B}$ with M samples as an example, to mitigate the issue of false-negative pairs, we propose to construct the reliable positive pairs set $\mathcal{P}_i^v$ and negative pairs set $\mathcal{N}_i^v$ as follows:

$$\mathcal{P}_i^v = \{j \mid \mathcal{G}_{i,j}^v = 1,\ \boldsymbol{Y}_{i,j} = 1, \quad \forall j \in [1, M]\}, \tag{9}$$

$$\mathcal{N}_i^v = \{j \mid \mathcal{G}_{i,j}^v = 0,\ \boldsymbol{Y}_{i,j} = 0, \quad \forall j \in [1, M]\}. \tag{10}$$

Based on the constructed reliable positive and negative pairs, we design the following contrastive loss for the RNGPA module (11), as shown at the bottom of the next page, where τ_I serves as the temperature parameter. $s(\cdot,\cdot)$ is also a similarity function and calculated as $s(\boldsymbol{z}_i, \boldsymbol{z}_j) = \frac{\boldsymbol{z}_i^{\mathrm{T}} \boldsymbol{z}_j}{\|\boldsymbol{z}_k\| \|\boldsymbol{z}_j\|}$. From Eq. (11), we can find that the constrastive loss of the RNGPA module can also be divided into two key components: *instance cohesion* and *instance separation*. The numerator represents the similarity between an instance and its positive pairs, encouraging closer representations. The denominator includes similarities with negative pairs, aiding in distinguishing instances. As shown in Fig. 3b, the instance cohesion term focuses on aligning positive pairs related to the consensus representation and view-specific representation. The instance separation term is designed to push the negative pairs of the consensus representation and the view-specific representation away.

D. Two-Stage Training Paradigm and Overall Objective Loss

We train our unsupervised deep multi-view clustering network in warm-up and fine-tuning stages. In the warm-up stage, we focus on training the view-specific autoencoders. In the fine-tuning stage, we aim to optimize the global network with the initialized parameters of these trained deep autoencoders.

1) Warm-up Training Stage: For our proposed network, each view is equipped with a view-specific autoencoder. Using randomly initialized parameters for these deep autoencoders may lead the model to converge to local optima during training. Therefore, we first train these deep autoencoders to obtain better parameters $\{\boldsymbol{\theta}^v\}_{v=1}^V$ and $\{\boldsymbol{\phi}^v\}_{v=1}^V$ to expedite the convergence of the model towards the optimal solution. Specifically, in the warm-up stage, the overall objective loss is to minimize the reconstruction loss Eq. (3) between the original sample $\boldsymbol{x}_i^v$ and its reconstructed counterpart $\hat{\boldsymbol{x}}_i^v$ as follows:

$$\mathcal{L}_{\mathrm{wu}} = \mathcal{L}_{\mathrm{rec}}. \tag{12}$$

2) Fine-Tuning Stage: In the fine-tuning stage, we exploit EM optimization strategy to train the network.

E-step: The purpose of this step is to estimate $p(k|\boldsymbol{z}_i)$ for the proposed DCD and RNGPA modules. Specifically, we perform the K-means algorithm on the obtained consensus representation to obtain a unique clustering result of the multi-view data.

M-step: In this step, we take into account the losses of all modules and optimize the following overall objective loss:

$$\mathcal{L}_{\mathrm{ft}} = \mathcal{L}_{\mathrm{rec}} + \alpha\mathcal{L}_{\mathrm{dcd}} + \beta\mathcal{L}_{\mathrm{rngpa}}, \tag{13}$$

where α and β are two hyper-parameters that control the balance among the three loss components.

The training procedure of the proposed method is outlined in Algorithm 1. The final clustering results are obtained by performing K-means clustering on the consensus representation $\boldsymbol{Z}$ produced by the adaptive feature fusion module.

Algorithm 1 Training Algorithm

Input: Multi-view Dataset $\mathbf{X}$;
The number of cluster K
Initialization
Initialize autoencoder parameters by minimizing $\mathcal{L}_{\mathrm{wu}}$ in Eq. (12)
repeat
E-step: update $\{p(k|\boldsymbol{z}_i)\}$ for each sample in $\mathbf{X}$ using K-means clustering
M-step: **repeat**
Randomly sample a mini-batch $\mathcal{B}$ from $\mathbf{X}$
for *each* $\boldsymbol{x}_i$ *in* $\mathcal{B}$ **do**
Compute clusters' representations using Eqs. (6) and (7)
$\mathcal{L}_{\mathrm{rec}} \leftarrow$ Eq. (3)
$\mathcal{L}_{\mathrm{dcd}} \leftarrow$ Eq. (5)
$\mathcal{L}_{\mathrm{rngpa}} \leftarrow$ Eq. (11)
end
$\mathcal{L}_{\mathrm{ft}} \leftarrow$ Eq. (13)
Update all the parameters by miniminzing $\mathcal{L}_{\mathrm{ft}}$ with Adam optimizer
until *an epoch finished*;
until *reaching max epochs*;
Output: The clustering results $\{p(k|\boldsymbol{z}_i)\}$

IV. Experiments

A. Datasets

The experiments are performed on the following publicly available datasets. Their detailed information is provided in Table II.

- *SentencesNYU v2 (RGB-D):* RGB-D [51] includes indoor scene images with descriptions. We use a ResNet-50 network pre-trained on ImageNet dataset to extract the

TABLE II
DESCRIPTIONS OF THE UTILIZED BENCHMARK DATASETS

Dataset	Sample	View	Class	Dimension
RGB-D	1449	2	13	2048/300
Cora	2708	4	7	2708/1433/2708/2708
CCV	6773	3	20	5000/5000/4000
Hdigit	10000	2	10	784/256
ALOI	10800	4	100	77/13/64/125
Digit-Product	30000	2	10	1024/1024

features from images as the first view. The second view is generated by a doc2vec model pre-trained on the Wikipedia dataset from the embedded image descriptions.

- *Cora:* Cora [52] comprises 2,708 documents categorized into seven classes. Four kinds of features are selected as four views, *i.e.*, content, inbound, outbound, and cites.
- *Columbia Consumer Video (CCV):* The video dataset CCV [53] containing 6,773 samples distributed across 20 classes, offers manually crafted Bag-of-Words representations as three views, such as STIP, SIFT, and MFCC.
- *Hdigit:* Hdigit [33] is derived from the MNIST and USPS handwritten digits datasets, comprising 10,000 samples and two distinct views.
- *ALOI:* ALOI [54] is a subset of ALOI-1k, where color similarity, Haralick, HSV, and RGB features are extracted from each image as its four view representations.
- *Digit-Product:* Similar to Hdigit, Digit-Product [29] sourced from both the MNIST and Fashion Handwritten digits datasets, encompasses 30,000 samples and two views.

B. Compared Methods and Evaluation Measures

We compare DCMVC against the following traditional and deep multi-view clustering methods.

- K-means: K-means [55], a classic clustering method, partitions data by minimizing distances between points and cluster centroids.
- BMVC: Binary Multi-View Clustering (BMVC) [56] incorporates two essential elements: the acquisition of a compacted collaborative discrete representation and the learning of a binary clustering structure.
- LMVSC: Large-scale Multi-View Subspace Clustering (LMVSC) constructs a smaller graph for each view between raw data points and anchors, followed by an integration mechanism to merge these graphs.
- FPMVS-CAG: Fast Parameter-free Multi-view Subspace Clustering with Consensus Anchor Guidance (FPMVS-CAG) [57] integrates anchor selection and subsequent subspace graph construction into a unified optimization process.
- EAMC: End-to-end Adversarial-attention network for Multi-modal Clustering (EAMC) [58] leverages adversarial learning to align latent feature distributions and employs attention mechanisms to quantify the importance of modalities.
- SiMVC: Simple Multi-view Clustering (SiMVC) [47] exhibits competitive or superior performance by prioritizing informative views through a learned linear combination mechanism.
- DSMVC: Deep Safe Multi-view Clustering (DSMVC) [59] automatically selects informative features, mitigating the risk of performance degradation caused by increasing views, and ensuring improved clustering performance in diverse scenarios.
- ProPos: Prototype scattering and Positive sampling (ProPos) [40] maximizes distances between prototypes to enhance representation uniformity and aligns augmented views with sampled neighbors for within-cluster compactness.
- CoMVC: Contrastive Multi-view Clustering (CoMVC) [47] combines SiMVC with a selective contrastive alignment module. It can effectively leverage alignment advantages and maintain the prioritization of informative views.
- MFLVC: Multi-level feature learning for contrastive multi-view clustering (MFLVC) [31] effectively balances the reconstruction of view-private information and the learning of common semantics.

$$
\begin{aligned}
\mathcal{L}_{\text{rngpa}} &= \frac{1}{M}\sum_{v=1}^{V}\sum_{i=1}^{M} -\log \frac{\exp\left(\dfrac{\sum\limits_{j\in\mathcal{P}_i^v} s\left(z_i, z_j^v\right)}{\tau_I}\right)}{\sum\limits_{j\in\mathcal{N}_i^v}\exp\left(\dfrac{s\left(z_i, z_j^v\right)}{\tau_I}\right) + \sum\limits_{j\in\mathcal{N}_i^v}\exp\left(\dfrac{s\left(z_i^v, z_j^v\right)}{\tau_I}\right)} \\
&= \underbrace{\frac{1}{M}\sum_{v=1}^{V}\sum_{i=1}^{M} -\frac{\sum\limits_{j\in\mathcal{P}_i^v} s\left(z_i, z_j^v\right)}{\tau_I}}_{\text{instance cohesion}} \\
&\quad + \underbrace{\frac{1}{M}\sum_{v=1}^{V}\sum_{i=1}^{M}\log\left(\sum_{j\in\mathcal{N}_i^v}\exp\left(\frac{s\left(z_i, z_j^v\right)}{\tau_I}\right) + \sum_{j\in\mathcal{N}_i^v}\exp\left(\frac{s\left(z_i^v, z_j^v\right)}{\tau_I}\right)\right)}_{\text{instance separation}},
\end{aligned}
\tag{11}
$$

TABLE III
CLUSTERING RESULTS ON RGB-D, CORA, AND CCV DATASETS

Dataset	RGB-D				Cora				CCV			
Evaluation metric	ACC	NMI	PUR	ARI	ACC	NMI	PUR	ARI	ACC	NMI	PUR	ARI
K-means	0.4044	0.3793	0.5328	0.2182	0.3674	0.1511	0.3818	0.0285	0.1992	0.1779	0.2257	0.0645
BMVC	0.2236	0.1191	0.3513	0.0563	0.2718	0.0698	0.3413	0.0301	0.1877	0.1651	0.2283	0.0390
LMVSC	0.4589	0.4008	0.4631	0.2613	0.3427	0.1336	**0.7105**	0.0277	0.1313	0.0859	**0.4261**	0.0097
FPMVS-CAG	0.4727	0.3722	0.5439	0.2486	0.4753	0.2167	0.4753	0.1826	0.2116	0.1643	0.2257	0.0574
EAMC	0.3982	0.3041	0.5073	0.2115	0.2315	0.0060	0.3028	0.0066	0.1168	0.0879	0.1391	0.0038
SiMVC	0.3527	0.3433	0.5176	0.1985	0.2700	0.0817	0.3597	0.0550	0.1598	0.1190	0.1864	0.0452
DSMVC	0.4458	0.3956	0.5707	0.2680	0.3083	0.0844	0.3648	0.0593	0.1626	0.1217	0.1939	0.0478
ProPos	0.3865	0.3702	0.5521	0.2317	0.5827	0.4525	0.6370	0.3574	0.2079	0.1830	0.2348	0.0676
CoMVC	0.3795	0.3523	0.5458	0.2029	0.2999	0.0883	0.3737	0.0600	0.2802	0.2921	0.3388	0.1336
MFLVC	0.4148	0.2194	0.4237	0.1782	0.2810	0.1310	0.3977	0.0637	0.3204	0.3161	0.3594	0.1599
GCFAgg	0.2622	0.2039	0.4113	0.1177	0.2622	0.1087	0.3549	0.0603	0.3399	0.3166	0.3728	0.1677
DCMVC (Ours)	**0.5245**	**0.4227**	**0.6128**	**0.3265**	**0.6691**	**0.4777**	0.6809	**0.4296**	**0.4026**	**0.3447**	0.4218	**0.1958**

- GCFAgg: Global and Cross-view Feature Aggregation (GCFAgg) [60] aligns consensus and view-specific representations through a structure-guided contrastive learning module.

K-means, as a single-view clustering method, its input is the concatenated features of all views. Among the other methods, BMVC, LMVSC, and FPMVS-CAG belong to shallow representation learning-based methods, while EAMC, DSMVC, SiMVC, CoMVC, MFLVC, and GCFAgg are deep representation learning-based methods. Since ProPos is a single-view clustering method, we treat the second view in the multi-view datasets as an augmentation of the first view, applying the same network architecture and training strategy as our method.

The selected quantitative metrics include unsupervised clustering accuracy (ACC) [61], normalized mutual information (NMI) [62], Purity (PUR) [63], adjusted rand index (ARI) [64], and F-score [65]. For these evaluation metrics, higher values indicate better performance.

C. Implementation Details

Since all views are represented by vector-based feature type in the above datasets, we adopt the fully connected (Fc) layers with general settings as the main layers in the proposed deep network. Specifically, for each view, we simply set the architecture of the encoder network as: Input$-$Fc$_{500}-$Fc$_{500}-$Fc$_{2000}-$Fc$_{256}$. The decoder has mirrored architecture as its corresponding encoder for each view. Moreover, the subsequent configurations are consistent across all experimental datasets. ReLU [66] serves as the activation function, and Adam [67] is selected as the optimizer with a default learning rate of 0.0001. The mini-batch size is 256. In the warm-up stage, all view-specific autoencoders are trained for 200 epochs. In the fine-tuning stage, the network is trained with 100 epochs on each dataset, and the temperature parameters τ_C and τ_I are fixed at 0.5. For datasets with fewer than 10,000 samples, we construct the nearest neighbor graph across the entire dataset in each view, selecting the number of nearest neighbors from the set $\{3, 5, 7, 10\}$. To enhance the computational efficiency on the datasets with more than 10,000 samples, we construct the nearest neighbor graph on the mini-batch data and select the number of nearest neighbors from the set $\{2, 3, 4, 5\}$. Additionally, for the two hyper-parameters α and β, their values are chosen from $\{1, 0.1, 0.01, 0.001, 0.0001, 0.00001\}$.

The experiments of DCMVC are conducted on an Ubuntu 22.04 platform. The hardware configuration includes an NVIDIA 3090 graphics processing unit (GPU), an Intel i7-11700 CPU, and 32 GB of RAM. To ensure an equitable comparison, the reported experimental results of the compared methods are obtained by implementing the open-source codes with corresponding suggested settings.

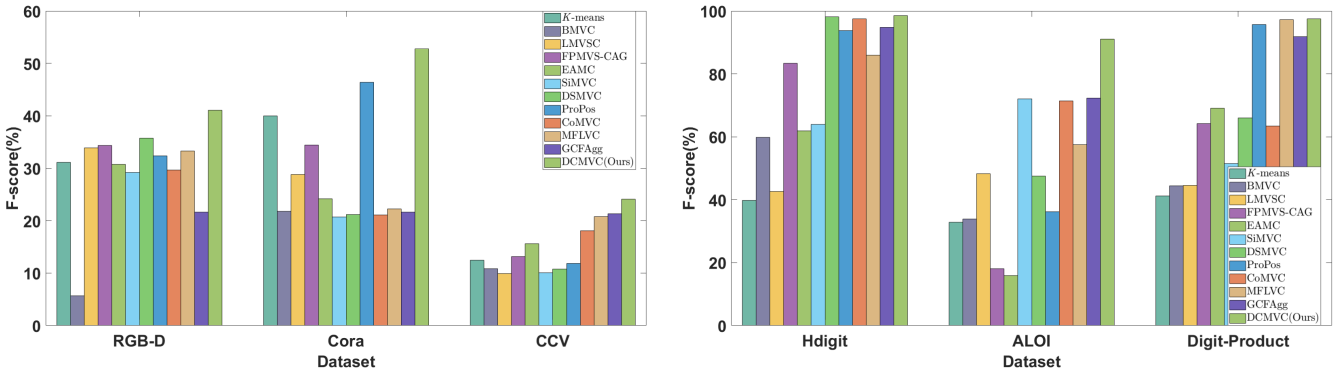
D. Experimental Results

The experimental results are outlined in Tables III, IV, and Fig. 4. The optimal and suboptimal results are highlighted in bold and underlined, respectively. It is noticeable that:

- In most cases, our DCMVC consistently outperforms other models in terms of quantitative metrics across all datasets. Simultaneously, DCMVC obtains a substantial performance improvement over the compared methods on RGB-D, CCV, Cora, and ALOI datasets. Compared to the suboptimal results in terms of ACC, it achieves enhancements of about 5.18%, 19.38%, 6.27%, and 17.69%, respectively. Compared with the other methods, the better performance of DCMVC indicates its superior capability in capturing and utilizing complementary information from different views.
- In general, single-view clustering (*i.e.*, K-means) exhibits inferior performance compared to multi-view methods. However, many compared MVC methods show limited performance, particularly on RGB-D and Cora datasets. For example, EAMC, SiMVC, and DSMVC perform worse than K-means methods on these datasets. This situation can be attributed to the fact that many MVC methods fail to extract discriminative representations from multiple views, thereby compromising the clustering performance. This observation suggests that simply having multiple views is not sufficient; the key lies in effectively integrating and utilizing the multi-view information. Our method employs dual contrast mechanisms, enabling the extraction of discriminative features from multiple views.
- DCMVC outperforms those contrastive-based methods, e.g., CoMVC, MFLVC, and GCFAgg. This demonstrates

TABLE IV
CLUSTERING RESULTS ON HDIGIT, ALOI, AND DIGIT-PRODUCT DATASETS

Dataset	Hdigit				ALOI				Digit-Product			
Evaluation metric	ACC	NMI	PUR	ARI	ACC	NMI	PUR	ARI	ACC	NMI	PUR	ARI
K-means	0.5291	0.4717	0.5473	0.3299	0.4614	0.6711	0.4950	0.3076	0.4998	0.4779	0.5685	0.3425
BMVC	0.7141	0.5950	0.7511	0.5527	0.5323	0.7192	0.5640	0.3298	0.5909	0.4294	0.6321	0.3815
LMVSC	0.5424	0.4837	0.5798	0.3612	0.5323	0.7192	0.5640	0.3298	0.5585	0.5013	0.6102	0.3818
FPMVS-CAG	0.9104	0.8164	0.9104	0.8148	0.3371	0.6476	0.3417	0.1661	0.7560	0.6884	0.7563	0.5989
EAMC	0.5288	0.7440	0.5902	0.5289	0.1508	0.5068	0.1799	0.0586	0.7553	0.7844	0.7733	0.6346
SiMVC	0.7119	0.6927	0.7125	0.5986	0.7401	0.9029	0.7534	0.7091	0.6015	0.5923	0.6313	0.4602
DSMVC	0.9909	0.9745	0.9909	0.9799	0.5487	0.7658	0.5600	0.4572	0.7656	0.6970	0.7656	0.6180
ProPos	0.9683	0.9179	0.9683	0.9310	0.4656	0.6719	0.4959	0.4959	0.9779	0.9458	0.9779	0.9521
CoMVC	0.9872	0.9656	0.9872	0.9718	0.7221	0.8997	0.7490	0.7035	0.7183	0.6802	0.7301	0.5923
MFLVC	0.9266	0.8425	0.9266	0.8434	0.5499	0.8490	0.5499	0.5259	0.9857	**0.9648**	0.9857	0.9688
GCFAgg	0.9738	0.9292	0.9738	0.9428	0.7631	0.8951	0.7867	0.7171	0.9509	0.9403	0.9509	0.9089
DCMVC (Ours)	**0.9928**	**0.9781**	**0.9928**	**0.9841**	**0.9400**	**0.9626**	**0.9438**	**0.9097**	**0.9874**	0.9640	**0.9874**	**0.9722**



(a) F-score performance on RGB-D, Cora and CCV datasets.

(b) F-score performance on Hdigit, ALOI and Digit-product datasets.

Fig. 4. F-score performance across all datasets.

that the dual contrast mechanism introduced in our method is beneficial to obtain more discriminative representations. In addition, a dynamic cluster diffusion module can produce distinct and well-separated clusters. At the instance-level, the reliable neighbor-guided positive alignment module can alleviate the negative impact of false-negative pairs and enhance the within-cluster compactness by introducing both the nearest neighbor information and pseudo-labels. By integrating these meaningful modules into a global optimization network, a discriminative representation with a clustering-friendly structure can be learned, and thus a better clustering performance is obtained. Compared to ProPos, our method effectively handles multi-view data, leading to superior performance across all the multi-view datasets. This comprehensive approach ensures that our method not only leverages the strengths of multiple views but also effectively addresses the issue of false-negative pairs, leading to more robust and reliable clustering outcomes.

E. Ablation Study

In this section, we conduct detailed ablation studies to gain deeper insights into the designed submodules for multi-view clustering. The ablation results are shown in Table V.

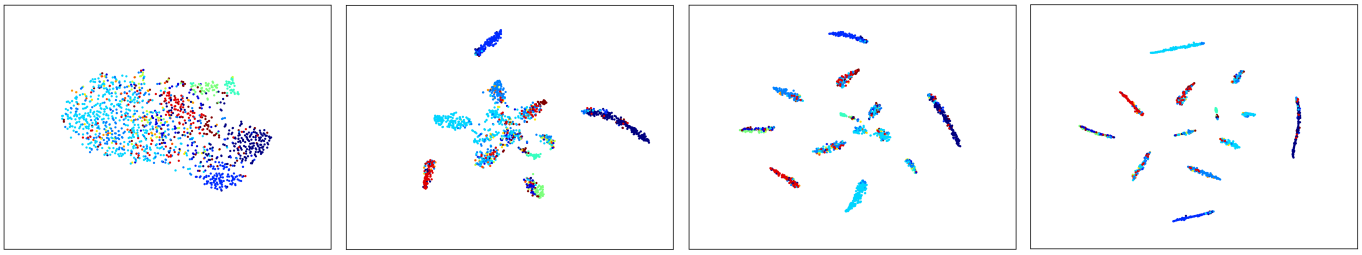
1) Effect of Warm-up Stage: We compare the clustering performance of DCMVC with its counterpart without the warm-up stage, referred to as DCMVC w/o warm-up. As shown in Table V, DCMVC w/o warm-up performs worse than DCMVC in terms of all evaluation metrics across three datasets. In particular, compared to DCMVC, the performance of DCMVC w/o warm-up declines by 11% and 8.93% in terms of ACC and PUR on the CCV dataset, respectively. This underscores the vital role of the warm-up stage in initializing parameters.

2) Effect of Adaptive Feature Fusion: To assess the effectiveness of the adaptive feature fusion approach introduced in our network, we conduct experiments to compare DCMVC and its degraded network that uses a simple average fusion, referred to as DCMVC w/o AFF. In other words, for DCMVC w/o AFF, we assign equal weights to each view. As depicted in Table V, the clustering performance of DCMVC slightly outperforms that of DCMVC w/o AFF. From these results, it can be observed that compared to average fusion, adaptive feature fusion enables the model to acquire more comprehensive and adaptive representations.

3) Effect of DCD and RNGPA: Similarly, we conduct experiments to evaluate the effectiveness of DCD and RNGPA modules by comparing the proposed method and its degraded networks without either one of the two modules, termed DCMVC w/o DCD) and DCMVC w/o RNGPA. From the experimental results shown in Table V, it can be noted that the absence of either DCD or RNGPA leads to a decrease in clustering performance. For example, on the ALOI dataset, the absence of DCD results in a 5.25% decrease in ACC. On the CCV dataset, the performance of DCMVC w/o RNGPA is

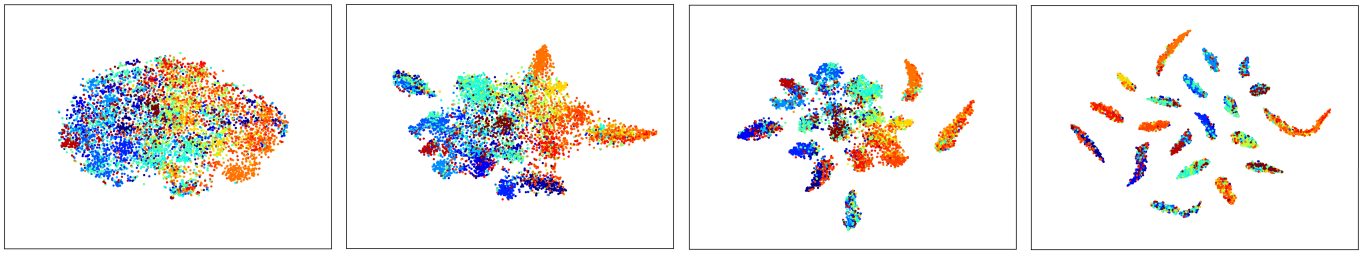
TABLE V
ABLATION STUDY FOCUSING ON THE KEY COMPONENTS

Method	RGB-D			CCV			ALOI		
	ACC	NMI	PUR	ACC	NMI	PUR	ACC	NMI	PUR
DCMVC w/o warm-up	0.5031	0.3857	0.6087	0.2926	0.2708	0.3325	0.9097	0.9522	0.9195
DCMVC w/o AAF	0.5045	0.4121	0.5977	0.3985	0.3428	0.4208	0.9206	0.9591	0.9297
DCMVC w/o DCD	0.5052	0.4166	0.6108	0.3842	0.3423	0.4078	0.8875	0.9438	0.9045
DCMVC w/o RNGPA	0.2885	0.2560	0.4852	0.2488	0.2452	0.2971	0.9308	0.9612	0.9376
DCMVC (Ours)	**0.5245**	**0.4227**	**0.6128**	**0.4026**	**0.3447**	**0.4218**	**0.9400**	**0.9626**	**0.9438**



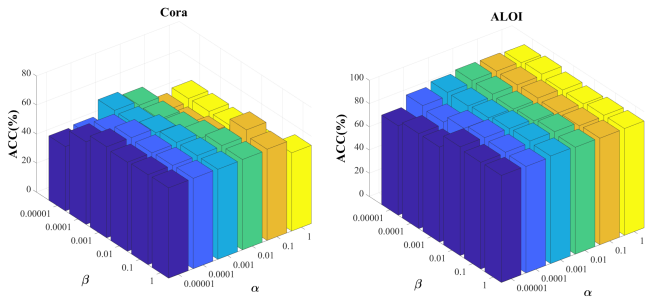
(a) Epoch 0 (b) Epoch 25 (c) Epoch 50 (d) Epoch 100

Fig. 5. Visualization of features in RGB-D for the dual contrastive learning process. The same color indicates features belonging to the same class.



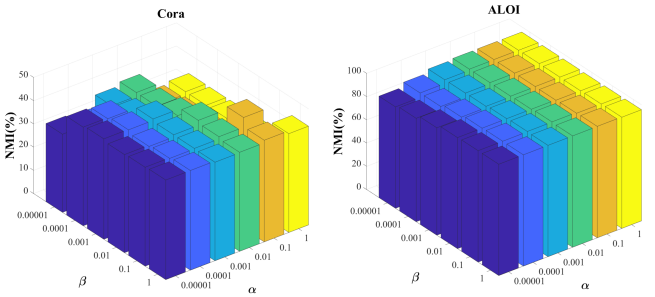
(a) Epoch 0 (b) Epoch 25 (c) Epoch 50 (d) Epoch 100

Fig. 6. Visualization of features in CCV for the dual contrastive learning process. The same color indicates features belonging to the same class.



(a) (b)

Fig. 7. The ACC values of DCMVC with different α and β combinations on two representative datasets.



(a) (b)

Fig. 8. The NMI values of DCMVC with different α and β combinations on two representative datasets.

about 15.38% lower than DCMVC in terms of ACC, which demonstrates that RNGPA module can effectively enhance the clustering performance of the designed network. Moreover, we can find that the best performance can be obtained by combining DCD and RNGPA modules simultaneously. This is mainly because the tight within-cluster compactness achieved by RNGPA can further ensure the effectiveness of DCD in pulling the cluster centers of different clusters together.

F. Model Analysis Based on Visualization

1) Visualization Analysis: To validate that our method can obtain the discriminative representation with a clustering-friendly structure, we conduct experiments on RGB-D and CCV datasets and use the t-SNE [68] method to visualize the learned consensus representations at some training steps. As depicted in Figs. 5 and 6, with the iterative training epoch increases, the clustering structure of the learned consensus representations gets clearer and clearer, where the distance interval between different clusters is getting larger and larger, and the distribution of data within clusters is becoming increasingly compacted.

2) Hyper-Parameter Analysis: We conduct experiments on two representative datasets, namely the Cora and ALOI datasets, to explore the sensitivity of hyper-parameters α, β, and the number of nearest neighbor samples γ. When performing sensitivity experiments for a single parameter, all

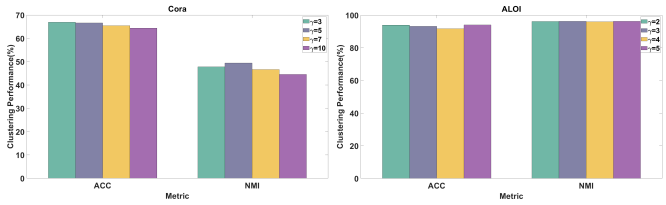


Fig. 9. The ACC and NMI values across various γ settings on two representative datasets.

other parameters are fixed at their optimal values. Figs. 7, 8 and 9 show the clustering performance of the DCMVC method across various combinations of α, β, and γ in terms of ACC and NMI. As seen in Figs. 7 and 8, the clustering performance of the DCMVC method on the Cora dataset exhibits significant fluctuations with varying combinations of α and β. In contrast, the clustering results on the ALOI dataset appear relatively stable. Observing Fig. 9, it becomes apparent that the clustering performance experiences a slight influence with respect to the number of nearest neighbor samples γ. In summary, our method exhibits remarkable robustness to hyper-parameter selections, maintaining consistent good performance across diverse settings.

V. Conclusion

In this paper, we proposed a new Dual Contrastive mechanism based deep Multi-View Clustering network (DCMVC). Compared with the existing MVC methods, DCMVC can learn clustering-friendly discriminative representations, in which different clusters are well-separated in the latent representation space and the data in the same cluster are distributed compactly. To fully consider the specific information of all distinct views, DCMVC introduces several view-specific auencoders to extract the view-specific features. Then, DCMVC introduces an adaptive representation fusion layer to learn the consensus representation. To ensure the clustering-friendly structure for the consensus representation, such as well-separated clusters and within-cluster compactness, two contrastive learning modules, *i.e.*, DCD and RNGPA, are integrated into the network. DCD module seeks to maximize inter-cluster distance by increasing the separation between clusters' representations in the consensus feature space. RNGPA module introduces a new reliable contrastive loss, which can improve the within-cluster compactness by fully exploring the pseudo-labels and nearest neighbor information to eliminate false-negative pairs. Experimental results demonstrate the capability of our method to learn discriminative representations and show that our method significantly surpasses current state-of-the-art methods in the multi-view clustering tasks.

In future work, we plan to extend our framework to address cross-modal matching and retrieval tasks. Another crucial direction is to develop more effective strategies for constructing positive and negative sample pairs to enhance contrastive learning. Furthermore, we will investigate improved clustering-friendly structures that better balance inter-cluster separation and within-cluster compactness. These efforts will contribute to the robustness and generalizability of our DCMVC framework in various applications.

References

[1] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic subspace clustering of high dimensional data for data mining applications," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Jun. 1998, pp. 94–105.

[2] R. M. Bommisetty, A. Khare, M. Khare, and P. Palanisamy, "Content-based video retrieval using integration of curvelet transform and simple linear iterative clustering," *Int. J. Image Graph.*, vol. 22, no. 2, Apr. 2022, Art. no. 2250018.

[3] Z. Zhang et al., "Flexible auto-weighted local-coordinate concept factorization: A robust framework for unsupervised clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 4, pp. 1523–1539, Apr. 2021.

[4] Z. Peng, H. Liu, Y. Jia, and J. Hou, "EGRC-net: Embedding-induced graph refinement clustering network," *IEEE Trans. Image Process.*, vol. 32, pp. 6457–6468, 2023.

[5] N. Dhanachandra, K. Manglem, and Y. J. Chanu, "Image segmentation using k -means clustering algorithm and subtractive clustering algorithm," *Proc. Comput. Sci.*, vol. 54, pp. 764–771, Jul. 2015.

[6] W. Kim, A. Kanezaki, and M. Tanaka, "Unsupervised learning of image segmentation based on differentiable feature clustering," *IEEE Trans. Image Process.*, vol. 29, pp. 8055–8068, 2020.

[7] Z. Zhang, L. Jia, M. Zhang, B. Li, L. Zhang, and F. Li, "Discriminative clustering on manifold for adaptive transductive classification," *Neural Netw.*, vol. 94, pp. 260–273, Oct. 2017.

[8] T. Hoya, "Reducing the number of centers in a probabilistic neural network via applying the first neighbor means clustering algorithm," *Array*, vol. 14, Jul. 2022, Art. no. 100161.

[9] G. Slavic, A. S. Alemaw, L. Marcenaro, D. Martín Gómez, and C. Regazzoni, "A Kalman variational autoencoder model assisted by odometric clustering for video frame prediction and anomaly detection," *IEEE Trans. Image Process.*, vol. 32, pp. 415–429, 2023.

[10] Z. Peng, H. Liu, Y. Jia, and J. Hou, "Adaptive attribute and structure subspace clustering network," *IEEE Trans. Image Process.*, vol. 31, pp. 3430–3439, 2022.

[11] R. A. Said and K. S. Mwitondi, "An integrated clustering method for pedagogical performance," *Array*, vol. 11, Sep. 2021, Art. no. 100064.

[12] J. Wen et al., "A survey on incomplete multiview clustering," *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 53, no. 2, pp. 1136–1149, Feb. 2023.

[13] G. Tzortzis and A. Likas, "Kernel-based weighted multi-view clustering," in *Proc. IEEE 12th Int. Conf. Data Mining*, Dec. 2012, pp. 675–684.

[14] X. Liu, Y. Dou, J. Yin, L. Wang, and E. Zhu, "Multiple kernel k-means clustering with matrix-induced regularization," in *Proc. AAAI Conf. Artif. Intell.*, 2016, vol. 30, no. 1, pp. 1–24.

[15] J. Liu, X. Liu, Y. Yang, Q. Liao, and Y. Xia, "Contrastive multi-view kernel learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 8, pp. 1–15, Jun. 2023.

[16] X. Zhu, S. Zhang, W. He, R. Hu, C. Lei, and P. Zhu, "One-step multi-view spectral clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 10, pp. 2022–2034, Oct. 2019.

[17] Q. Gao, W. Xia, Z. Wan, D. Xie, and P. Zhang, "Tensor-SVD based graph learning for multi-view subspace clustering," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, 2020, pp. 3930–3937.

[18] P. Zhang et al., "Consensus one-step multi-view subspace clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 10, pp. 4676–4689, Oct. 2022.

[19] Y. Jia, H. Liu, J. Hou, S. Kwong, and Q. Zhang, "Multi-view spectral clustering tailored tensor low-rank representation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 12, pp. 4784–4797, Dec. 2021.

[20] L. Li and H. He, "Bipartite graph based multi-view clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 7, pp. 3111–3125, Jul. 2022.

[21] C. Tang et al., "CGD: Multi-view clustering via cross-view graph diffusion," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 4, 2020, pp. 5924–5931.

[22] K. Zhan, F. Nie, J. Wang, and Y. Yang, "Multiview consensus graph clustering," *IEEE Trans. Image Process.*, vol. 28, no. 3, pp. 1261–1270, Mar. 2019.

[23] C. Zhang, Y. Liu, and H. Fu, "AE2-Nets: Autoencoder in autoencoder networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2577–2585.

[24] J. Wen et al., "DIMC-Net: Deep incomplete multi-view clustering network," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 3753–3761.

[25] C. Liu, S. Wu, R. Li, D. Jiang, and H.-S. Wong, "Self-supervised graph completion for incomplete multi-view clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 9, pp. 1–14, May 2023.

[26] J. Xu et al., "Adaptive feature projection with distribution alignment for deep incomplete multi-view clustering," *IEEE Trans. Image Process.*, vol. 32, pp. 1354–1366, 2023.

[27] P. Hu et al., "Deep supervised multi-view learning with graph priors," *IEEE Trans. Image Process.*, vol. 33, pp. 123–133, 2024.

[28] M. Yin, W. Huang, and J. Gao, "Shared generative latent representation learning for multi-view clustering," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 6688–6695.

[29] J. Xu et al., "Multi-VAE: Learning disentangled view-common and view-peculiar visual representations for multi-view clustering," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9234–9243.

[30] Q. Wang, Z. Tao, Q. Gao, and L. Jiao, "Multi-view subspace clustering via structured multi-pathway network," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 5, pp. 1–7, Jul. 2022.

[31] J. Xu, H. Tang, Y. Ren, L. Peng, X. Zhu, and L. He, "Multi-level feature learning for contrastive multi-view clustering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 16051–16060.

[32] S. Hu, G. Zou, C. Zhang, Z. Lou, R. Geng, and Y. Ye, "Joint contrastive triple-learning for deep multi-view clustering," *Inf. Process. Manage.*, vol. 60, no. 3, May 2023, Art. no. 103284.

[33] M.-S. Chen et al., "Representation learning in multi-view clustering: A literature review," *Data Sci. Eng.*, vol. 7, no. 3, pp. 225–241, Sep. 2022.

[34] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.

[35] G. Zhang, Z. Hu, G. Wen, J. Ma, and X. Zhu, "Dynamic graph convolutional networks by semi-supervised contrastive learning," *Pattern Recognit.*, vol. 139, Jul. 2023, Art. no. 109486.

[36] Z. Li, J. Wang, L. Hua, H. Liu, and W. Song, "Automatic tracking method for 3D human motion pose using contrastive learning," *Int. J. Image Graph.*, vol. 24, no. 3, May 2024, Art. no. 2550037.

[37] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. 37th Int. Conf. Mach. Learn.*, vol. 119, 2020, pp. 1597–1607.

[38] J.-B. Grill et al., "Bootstrap your own latent-a new approach to self-supervised learning," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst.*, 2020, pp. 21271–21284.

[39] Y. Li, P. Hu, Z. Liu, D. Peng, J. T. Zhou, and X. Peng, "Contrastive clustering," in *Proc. AAAI Conf. Artif. Intell.*, May 2021, vol. 35, no. 10, pp. 8547–8555.

[40] Z. Huang, J. Chen, J. Zhang, and H. Shan, "Learning representation for clustering via prototype scattering and positive sampling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 6, pp. 7509–7524, Jun. 2023.

[41] J. Qi, Y. Jia, H. Liu, and J. Hou, "Superpixel graph contrastive clustering with semantic-invariant augmentations for hyperspectral images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 1, no. 1, pp. 1–11, Aug. 2024.

[42] Y. Wang, X. Lin, L. Wu, W. Zhang, Q. Zhang, and X. Huang, "Robust subspace clustering for multi-view data by exploiting correlation consensus," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3939–3949, Nov. 2015.

[43] H. Wang, Y. Yang, and B. Liu, "GMC: Graph-based multi-view clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 6, pp. 1116–1129, Jun. 2020.

[44] Q. Wang, Z. Tao, W. Xia, Q. Gao, X. Cao, and L. Jiao, "Adversarial multiview clustering networks with adaptive fusion," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 10, pp. 1–13, Jun. 2022.

[45] C. Liu et al., "Self-guided partial graph propagation for incomplete multiview clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 1, no. 1, pp. 1–14, May 2023.

[46] W. Xia, S. Wang, M. Yang, Q. Gao, J. Han, and X. Gao, "Multi-view graph embedding clustering network: Joint self-supervision and block diagonal representation," *Neural Netw.*, vol. 145, pp. 1–9, Jan. 2022.

[47] D. J. Trosten, S. Løkse, R. Jenssen, and M. Kampffmeyer, "Reconsidering representation alignment for multi-view clustering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 1255–1265.

[48] J. Chen, H. Mao, W. L. Woo, and X. Peng, "Deep multiview clustering by contrasting cluster assignments," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 16752–16761.

[49] J. Wen et al., "Deep double incomplete multi-view multi-label learning with incomplete labels and missing views," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 1, no. 1, pp. 1–13, May 2023.

[50] C.-H. Yeh, C.-Y. Hong, Y.-C. Hsu, T.-L. Liu, Y. Chen, and Y. LeCun, "Decoupled contrastive learning," in *Proc. Eur. Conf. Comput. Vis.*, Oct. 2022, pp. 668–684.

[51] C. Kong, D. Lin, M. Bansal, R. Urtasun, and S. Fidler, "What are you talking about? Text-to-image coreference," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3558–3565.

[52] S.-G. Fang, D. Huang, X.-S. Cai, C.-D. Wang, C. He, and Y. Tang, "Efficient multi-view clustering via unified and discrete bipartite graph learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 1, no. 1, pp. 1–12, Jul. 2023.

[53] Y.-G. Jiang, G. Ye, S.-F. Chang, D. Ellis, and A. C. Loui, "Consumer video understanding: A benchmark database and an evaluation of human and machine performance," in *Proc. 1st ACM Int. Conf. Multimedia Retr.*, Apr. 2011, pp. 1–8.

[54] G.-Y. Zhang, D. Huang, and C.-D. Wang, "Facilitated low-rank multi-view subspace clustering," *Knowl.-Based Syst.*, vol. 260, Jan. 2023, Art. no. 110141.

[55] J. MacQueen et al., "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Math. Statist. Probab.*, vol. 1, 1967, pp. 281–297.

[56] Z. Zhang, L. Liu, F. Shen, H. T. Shen, and L. Shao, "Binary multi-view clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1774–1782, Jul. 2019.

[57] S. Wang et al., "Fast parameter-free multi-view subspace clustering with consensus anchor guidance," *IEEE Trans. Image Process.*, vol. 31, pp. 556–568, 2022.

[58] R. Zhou and Y.-D. Shen, "End-to-end adversarial-attention network for multi-modal clustering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 14619–14628.

[59] H. Tang and Y. Liu, "Deep safe multi-view clustering: Reducing the risk of clustering performance degradation caused by view increase," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, Jun. 2022, pp. 202–211.

[60] W. Yan et al., "GCFAgg: Global and cross-view feature aggregation for multi-view clustering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 19863–19872.

[61] W. Xu, X. Liu, and Y. Gong, "Document clustering based on non-negative matrix factorization," in *Proc. 26th Annu. Int. ACM SIGIR Conf. Res. Develop. Informaion Retr.*, Jul. 2003, pp. 267–273.

[62] A. Strehl and J. Ghosh, "Cluster ensembles—A knowledge reuse framework for combining multiple partitions," *J. Mach. Learn. Res.*, vol. 3, pp. 583–617, Dec. 2002.

[63] H. Schütze, C. D. Manning, and P. Raghavan, *Introduction To Information Retrieval*, vol. 39. Cambridge, U.K.: Cambridge Univ. Press, 2008.

[64] L. Hubert and P. Arabie, "Comparing partitions," *J. Classification*, vol. 2, no. 1, pp. 193–218, Dec. 1985.

[65] D. M. W. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation," 2020, *arXiv:2010.16061*.

[66] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, pp. 315–323.

[67] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent.*, 2015, pp. 1–24.

[68] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 1–24, 2008.