# Ambiguous Instance-Aware Contrastive Network with Multi-Level Matching for Multi-View Document Clustering

**Zhenqiu Shu, Teng Sun, Yunwei Luo , Zhengtao Yu***

Faculty of Information Engineering and Automation, Kunming University of Science and Technology
shuzhenqiu@163.com, st251100@163.com, lywinaaa@gmail.com, ztyu@hotmail.com

## Abstract

Multi-view document clustering (MvDC) aims to improve the accuracy and robustness of clustering by fully considering the complementarity of different views. However, in real-world clustering applications, most existing works suffer from the following challenges: 1) They primarily align multi-view data based on a single perspective, such as features and classes, thus ignoring the diversity and comprehensiveness of representations. 2) They treat each instance equally in cross-view contrastive learning without considering ambiguous ones, which weakens the model's discriminative ability. To address these problems, we propose an ambiguous instance-aware contrastive network with multi-level matching (AICN-MLM) for MvDC tasks. This model contains two key modules: a multi-level matching module and an ambiguous instance-aware contrastive learning module. The former attempts to align multi-view data from different perspectives, including features, pseudo-labels, and prototypes. The latter dynamically adjusts instance weights through a weight modulation function to highlight ambiguous instance pairs. Thus, our proposed method can effectively explore the consistency of multi-view document data and focus on ambiguous instances to enhance the model's discriminative ability. Extensive experimental results on several multi-view document datasets verify the effectiveness of our proposed method.

## Introduction

In this era of rapid information development, data often originates from various sources, such as news organizations reporting the same event in different languages. Many studies treat each language document as a separate view, utilizing multi-view analysis techniques to explore semantic information from multilingual documents. Compared with single-view data, multi-view data offers a more comprehensive understanding and usually achieves better results in downstream tasks. Multi-view clustering (MVC) (Chao, Sun, and Bi 2021) attempts to reveal hidden data patterns and group structures by integrating information from multi-view data. Existing MVC methods are divided into two offers: traditional MVC methods and deep MVC methods.

Traditional MVC methods (Wang et al. 2022b; Li et al. 2023; Shu et al. 2023) usually use shallow machine learning techniques to cluster multi-view data. For example, matrix factorization-based MVC methods (Zhao, Ding, and Fu 2017; Wang, Zhang, and Gao 2018) learn latent representations by decomposing the original multi-view data matrix. Graph-based MVC methods (Li et al. 2021; Huang et al. 2021) exploit the structural information of multi-view data by learning graph representations and then performing spectral clustering algorithms. Subspace-based MVC methods (Li et al. 2019a; Yin, Wu, and Wang 2015) aim to learn a low-dimensional subspace from original high-dimensional data that can better reveal the intrinsic structure. However, due to the limited representation ability of shallow learning methods, they usually perform poorly in complex scenarios such as multilingual document clustering.

Recently, deep MVC (DMVC) technology has attracted increasing attention due to the excellent feature representation ability of deep neural networks (DNNs). DMVC methods based on shared representation learning (Du et al. 2021; Xu et al. 2021a) learn nonlinear feature representations between different views by using autoencoders while maintaining consistency in the latent space through parameter sharing and reconstruction loss. DMVC methods based on self-supervised learning (Xu et al. 2022a; Xia et al. 2021) usually design self-supervised losses to optimize the autoencoder network jointly. GCN-based DMVC methods (Zhao, Yang, and Nie 2023; Cheng et al. 2021) use GCN to automatically learn node feature representations, thus effectively capturing high-order structural relationships between multi-view data. GAN-based DMVC algorithms (Li et al. 2019b; Shu et al. 2024) try to align the feature distributions of multi-view data through generators and discriminators. Most existing DMVC methods utilize contrastive learning techniques to explore the consistency of multi-view data by maximizing the similarity between positive pairs and minimizing it between negative pairs in the latent space (Xu et al. 2022b; Chen et al. 2023). However, they treat all positive and negative instance pairs equally, without considering that some instance pairs are easier to classify while others are more challenging and ambiguous.

Although the effectiveness of these techniques has been demonstrated in several real-world applications, they suffer from the following challenges: 1) The distribution of multi-

*Corresponding author

view data may be biased due to its multi-source attributes. Moreover, most existing methods only align multi-view data from the instance or cluster level. Therefore, they cannot provide a comprehensive alignment strategy across different views; 2) Previous contrastive learning approaches treat all instances equally and overlook ambiguous instances that are difficult to classify, thus limiting the model's discriminative ability.

To address the above issues, we propose a novel multi-view document clustering method, called ambiguous instance-aware contrastive network with multi-level matching (AICN-MLM). It includes three main modules: a multi-view data reconstruction module, a multi-level matching module, and an ambiguous instance-aware contrastive learning module. Specifically, the multi-view data reconstruction module aims to learn suitable feature representations for clustering from the original multi-view data. The multi-level matching module consists of multi-view similarity distribution matching (SDM) and cross-view prototype matching (CVPM). The SDM seeks to align multi-view data by minimizing the difference between the normalized multi-view similarity score distribution and the normalized pseudo-label matching distribution. The CVPM establishes the corresponding relationship by minimizing the Jensen-Shannon divergence (JS) between the prototypes of multi-view data. Additionally, the ambiguous instance-aware contrastive learning module uses an adaptive instance weighting function to dynamically adjust the weights of instance pairs according to their confidence levels. Therefore, it can pay more attention to ambiguous instance pairs and significantly improve the network's discriminative ability. Extensive experimental results on several datasets have shown the advantage of the proposed method in MvDC tasks.

The main contributions of this paper are summarized as follows:

- In our proposed method, we introduce a novel ambiguous instance-aware contrastive network that dynamically adjusts instance weights to emphasize ambiguous pairs, thereby enhancing the network's discriminative ability.
- We propose a multi-level matching strategy that aligns the feature of multi-view document data with pseudo-label distributions and view prototype sets, effectively exploring the consistency of multi-view document data.
- Extensive experiments are conducted on seven self-constructed multilingual document datasets and one public multilingual document dataset, demonstrating the superiority of our proposed method in multi-view document clustering.

## Related Work

### Deep Multi-view Clustering

In the past few years, DMVC has become an essential technique in multi-view clustering tasks, effectively extracting feature information from high-dimensional and complex multi-view data due to the powerful feature representation ability of DNNs. Among them, autoencoders are widely used to learn latent representations of multi-view data. Li et al. utilized autoencoders to learn latent shared representations between different views and adopted adversarial training to capture the latent distribution of multi-view data (Li et al. 2019b). Kusner et al. extracted the feature information of the multi-view data using variational autoencoders to model the data distribution and learn robust representations effectively (Kusner, Paige, and Hernández-Lobato 2017). Xu et al. learned to disentangle view-shared and view-specific visual representations and then achieved multi-view data clustering tasks (Xu et al. 2021b). Another type of DMVC method has recently utilized GCN to model the structural relationship between different views. Cheng et al. introduced a multi-view attribute graph convolutional network (MAGCN) to cluster multi-view attribute graph data (Cheng et al. 2021). Zhou et al. adopted adversarial learning and attention mechanisms to align latent feature distributions and quantify the importance of different modalities (Zhou and Shen 2020). Wang et al. combined adversarial training and adaptive fusion techniques to extract consistent latent representations from multi-view data (Wang et al. 2022a). However, existing DMVC methods only align multi-view data from a single perspective, ignoring the diversity and complementarity of multi-view data.

### Contrastive Learning

Contrastive learning (Chuang et al. 2020; Lin et al. 2021; Xu et al. 2022b; Yang et al. 2023) has been widely applied in deep clustering and representation learning. Its core goal is to maximize the similarity between positive instance pairs, minimize the similarity between negative instance pairs, and enhance the consistency between different views by aligning the encoded representations.

In recent years, various studies have explored different contrastive learning frameworks in MVC tasks (Xu et al. 2022b; Chen et al. 2023). Yang et al. proposed a dual contrastive calibration mechanism to maintain the consistency of similar but different instances in cross-view scenarios (Yang et al. 2023). Yan et al. attempted to learn consensus and view-specific representations from multiple views through global and cross-view feature aggregation, and then used structure-guided contrastive learning to align the feature representations of each view (Yan et al. 2023). These studies investigate the application of contrastive learning to enhance multi-view clustering performance. However, they treat all positive and negative instance pairs equally and neglect ambiguous ones, resulting in suboptimal performance improvement.

## Method

### Network Architecture

As shown in Figure 1, the proposed AIACN-MLM method consists of three modules: a multi-view data reconstruction module, a multi-level matching module, and an ambiguous instance-aware contrastive learning module. Next, we will provide a detailed introduction to these three modules.

### Multi-view Data Reconstruction

Since multi-view data usually contain redundant information and random noise that can adversely affect cluster-

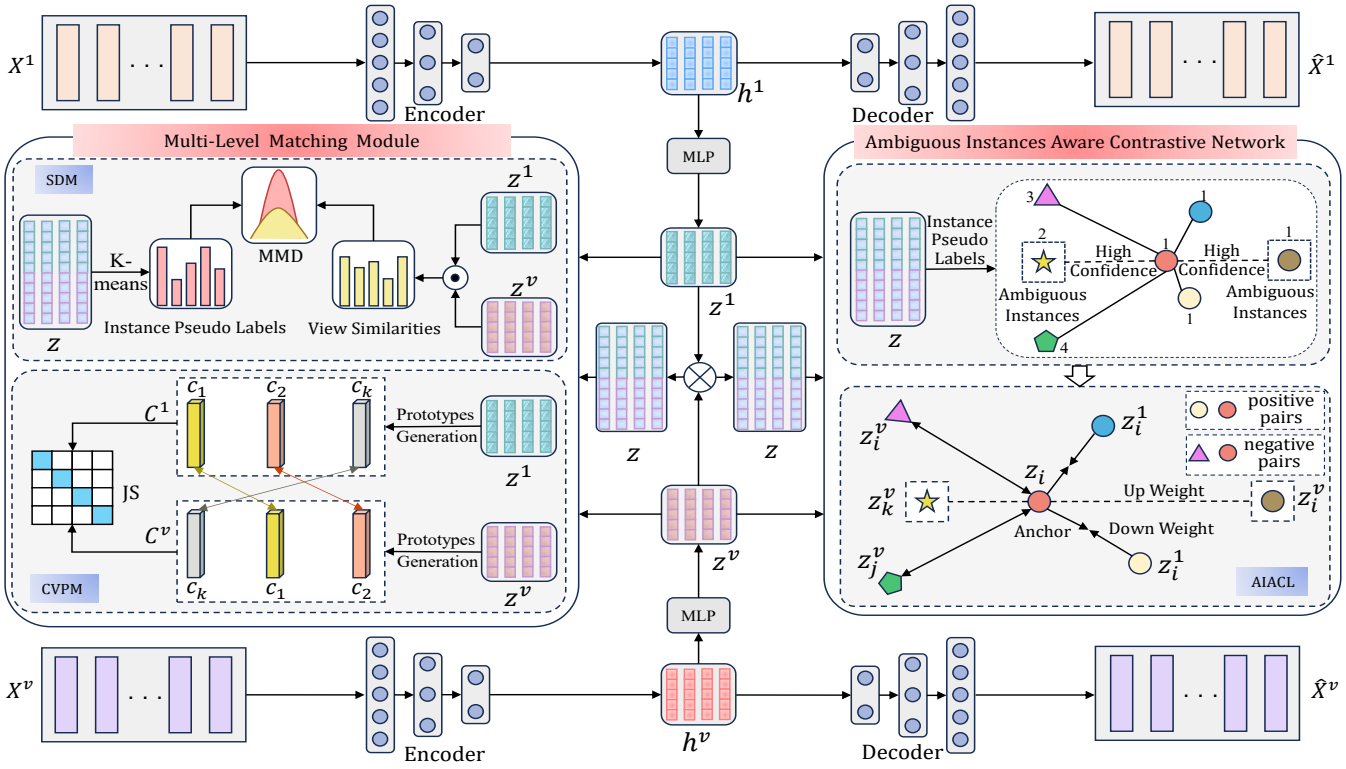



Figure 1: The overall structure of the proposed AICN-MLM method.

ing tasks, autoencoders are commonly used to learn salient representations from original multi-view data. Specifically, we denote the encoder and decoder of the $v$-th view as $f^v(x_i^v;\theta^v)$ and $g^v(x_i^v;\varphi^v)$, respectively, where $\theta^v$ and $\varphi^v$ represent the network parameters corresponding to the encoder and decoder, respectively. Thus, the low-dimensional feature learned from multi-view data using the model's encoder can be represented as follows:

$$h_i^v = f^v(x_i^v;\theta^v), \tag{1}$$

where $h_i^v$ is the embedded feature of $x_i^v$. The decoder reconstructs the instance by low-dimensional feature representation $h_i^v$. Therefore, multi-view data reconstruction using the decoder $g^v(h_i^v;\phi^v)$ can be expressed as follows:

$$\hat{x}_i^v = g^v(h_i^v;\phi^v). \tag{2}$$

where $\hat{x}_i^v$ denotes the reconstructed instance. The reconstruction loss from input $X^v$ to output $\hat{X}^v$ is denoted as $\mathcal{L}_{rec}$. Thus the reconstruction objective loss of all views is formulated as follows:

$$\begin{aligned}\mathcal{L}_{rec} &= \sum_{v=1}^{V}\left\|X^v - \hat{X}^v\right\|_F^2 \\ &= \sum_{v=1}^{V}\sum_{i=1}^{N}\|x_i^v - g^v(f^v(x_i^v;\theta^v);\phi^v)\|_2^2,\end{aligned} \tag{3}$$

where $N$ denotes the number of instances in a batch, and $V$ represents the number of views. By optimizing Eq.(3), we can obtain low-dimensional feature representations of multi-view data for clustering tasks.

## Multi-level Matching

To explore the consistency of multi-view data, we design a multi-level matching strategy in the proposed method. It includes the multi-view similarity distribution matching module (SDM) and the cross-view prototype matching module (CVPM). The SDM module aligns feature representations from different views by minimizing the Maximum Mean Discrepancy (MMD) between the cosine similarity distributions and pseudo label distributions. The CVPM module establishes prototype correspondences by optimizing the Jensen-Shannon divergence (JS) between cross-view prototypes.

In SDM, we initially stack a single-layer linear MLP on the low-level features $h^v$ to obtain the high-level features $z^v$, denoted as $z^v = F(\{h^v\}_{v=1}^V; W_H)$, where $W_H$ represents a set of learnable parameters. Next, we calculate the cosine similarity matrix $s$ of the high-level features $z^v$ of multi-view data. Then, the high-level features $z^v$ of each view are concatenated to obtain the common features $z$. We perform $k$-means clustering on $z$ to obtain the pseudo-labels. Finally, we minimize the MMD to align the view similarity distribution and the pseudo-labels distribution.

Given $N$ multi-view instance pairs, the high-level features of the $i$-th instance in the $m$-th view are denoted as $z_i^m$. Then we construct a set of representation pairs of different views, i.e. $\{(z_i^m, z_j^n), y_{i,j}\}_{i,j=1}^N$, where $y_{i,j}$ is the matching label. Specifically, $y_{i,j}$ indicates whether the $i$-th instance and the $j$-th instance belong to the same category. If $y_{i,j} = 1$, then $(z_i^m, z_i^n)$ is a matching pair of the same class, while $y_{i,j} = 0$

represents an unmatched pair. The probability of a matching pair can be calculated using the following softmax function:

$$p_{i,j} = \frac{\exp(sim(z_i^m, z_j^n)/\tau)}{\sum_{i,j=1}^{N} \exp(sim(z_i^m, z_j^n)/\tau)}, \tag{4}$$

where $sim(z_i^m, z_j^n) = \frac{z_i^{m\top} z_j^n}{\|z_i^m\| \|z_j^n\|}$ represents the cosine similarity between $z_i^m$ and $z_j^n$, and $\tau$ is the temperature hyperparameter that controls the sharpness of the probability distribution.

The loss of the SDM between the $m$-th view and the $n$-th view in a batch is calculated as follows:

$$\begin{aligned} \mathcal{L}_{sdm} &= \sum_{i=1}^{N} \sum_{j=1}^{N} d_{\mathcal{H}}(q_{i,j}; p_{i,j}) \\ &= \sum_{i=1}^{N} \sum_{j=1}^{N} \left\| \mathbb{E}_{q_{i,j}}[\varphi(q_{i,j})] - \mathbb{E}_{p_{i,j}}[\varphi(p_{i,j})] \right\|_{\mathcal{H}}^2, \end{aligned} \tag{5}$$

where $q_{i,j} = \frac{y_{i,j}}{\sum_{i,j=1}^{N} y_{i,j}}$ is the predict match probability. Here, $\mathcal{H}$ represents the Reproducing Kernel Hilbert Space (RKHS), and $\varphi(\cdot)$ represents the mapping function that projects the original feature to RKHS, such as the Gaussian kernel function.

We assume that the cluster distributions of two related views should be closer to each other. To this end, we introduce a prototype matrix $C = [c_1, c_2, \ldots, c_k] \in \mathbb{R}^{D \times K}$, where $K$ is the number of clusters, and $D$ is the dimension of the embedding across all views. Each $c_k$ represents a trainable prototype vector, indicating the center of the corresponding cluster. In practice, we use a linear layer to learn the prototype matrix $C$.

The cross-view prototype matching module uses Jensen-Shannon (JS) divergence to measure the pairwise prototype differences between cross-views in the feature space. Specifically, the loss function of the cross-view prototype matching can be defined as follows:

$$\begin{aligned} \mathcal{L}_{cvpm} =& \frac{1}{2} \sum_{k=1}^{K} p(C_k^m) \log \left( \frac{2 * p(C_k^m)}{p(C_k^m) + p(C_k^n)} \right) + \\ & \frac{1}{2} \sum_{k=1}^{K} p(C_k^n) \log \left( \frac{2 * p(C_k^n)}{p(C_k^n) + p(C_k^m)} \right), \end{aligned} \tag{6}$$

where $p(C_k^m)$ and $p(C_k^n)$ represent the probability distribution of the $k$-th prototype in the $m$-th view and the $n$-th view, respectively.

By matching the prototype-to-prototype correspondence between each pair of views, this module calibrates the relationship between prototypes in different views, thereby solving the prototype-shift problem and further improving clustering performance.

## Ambiguous Instance Aware Contrastive Learning

Traditional contrastive learning methods minimize InfoNCE loss by pulling together instances of the same category from different views while pushing dissimilar instances apart. However, these methods treat ambiguous and clear instance pairs equally, which can limit the discriminative ability of the network. To address this issue, we propose a weight adjustment function $\mathcal{M}$ to dynamically adjust the weights of instance pairs during training. Thus, we first calculate the distance from the instance to the cluster center, and then obtain the confidence score. The top $\lambda$ instances, based on this score, form the high-confidence instance set $H \in \mathbb{R}^M$. Here, $\lambda$ is the confidence hyperparameter, and $M$ is the number of high-confidence instances. We then derive the pair pseudo-labels $Q \in \mathbb{R}^{K \times N}$ from the instance pseudo-labels $P$ as follows:

$$Q_{ij} = \begin{cases} 1 & P_i = P_j, \\ 0 & P_i \neq P_j. \end{cases} \tag{7}$$

Based on the similarity function $S$ and the pair pseudo-labels $Q$, the weight adjustment function $\mathcal{M}$ is formulated as follows:

$$\mathcal{M}(z_i^m, z_j^n) = \begin{cases} 1 & \text{if } i, j \notin H, \\ |Q_{ij} - Norm(S(z_i^m, z_j^n))|^{\gamma} & \text{otherwise}, \end{cases} \tag{8}$$

where $z_i^m$ represents the $i$-th instance of the $m$-th view, $\gamma$ is the focusing factor, and $Norm$ represents the Min-Max normalization. $S(z_i^m, z_j^n)$ represents the cosine similarity between the $i$-th instance in $m$-th view and the $j$-th instance in $n$-th view. In Eq.(8), when the confidence of the instance is low, we keep the initial setting in the InfoNCE loss. When the instance has high confidence, the instance weight is modulated by the pseudo information and instance similarity. $\mathcal{M}$ can increase the weight of ambiguous instances while reducing the weight of clear instances.

Specifically, when the $i$-th and $j$-th instances are identified as a positive pair ($Q_{ij} = 1$), their possibility of being classified into the same category increases with their similarity. Therefore, $\mathcal{M}$ increases the weight of positive pairs with less similarity (ambiguous instances) and decreases the weight of positive pairs with greater similarity (clear instances). The ambiguous instance-aware contrastive loss for the $i$-th instance of the $m$-th view is given as follows:

$$\begin{aligned} &\mathcal{L}(z_i^m) = -\log \\ &\frac{\sum_{i=1}^{N} e^{\mathcal{M}(z_i^m, z_i^n) \cdot S(z_i^m, z_i^n)}}{\sum_{j=1}^{N} (e^{\mathcal{M}(z_i^m, z_j^m) \cdot S(z_i^m, z_j^m)} + e^{\mathcal{M}(z_i^m, z_j^n) \cdot S(z_i^m, z_j^n)})}. \end{aligned} \tag{9}$$

In contrast to traditional InfoNCE loss, our weight modulation function $\mathcal{M}$ increases the weight of ambiguous instance pairs and reduces the weight of clear instance pairs. The overall loss formula is given as follows:

$$\mathcal{L}_{aicl} = \frac{1}{N} \frac{1}{V} \sum_{m=1}^{V} \sum_{i=1}^{N} \mathcal{L}(z_i^m). \tag{10}$$

This ambiguous instance-aware contrastive loss can guide the network to pay more attention to ambiguous instances, thereby enhancing the model's discriminative ability.

## Loss Function

By integrating the reconstruction loss $\mathcal{L}_{rec}$, the similarity distribution matching loss $\mathcal{L}_{sdm}$, the cross-view prototype

matching loss $\mathcal{L}_{cvpm}$ and the ambiguous instance-aware contrastive loss $\mathcal{L}_{aicl}$, the total loss function of the proposed method is formulated as follows:

$$\mathcal{L} = \mathcal{L}_{rec} + \mathcal{L}_{aicl} + \alpha\mathcal{L}_{cvpm} + \beta\mathcal{L}_{sdm}, \tag{11}$$

where $\alpha$ and $\beta$ are the trade-off hyperparameters.

# Experiments

In this section, we conducted extensive experiments to evaluate the effectiveness of the proposed method in MVC tasks.

## Datasets

The experiments were carried out using two datasets: a self-constructed multilingual document dataset (**KUST**) and a public multilingual document dataset (**Reuters**).

The **KUST** multi-view document dataset includes the following seven subsets: (1) **KUST-ETD** has 10,000 instances spanning 10 themes, with English and Thai document views. (2) **KUST-CTD** consists of 10,000 instances from 10 themes, each with views of Chinese and Thai documents. (3) **KUST-CVD** comprises 10,000 instances with Chinese and Vietnamese documents, divided into 10 distinct classes. (4) **KUST-CED** contains 10,000 instances with Chinese and English documents, categorized into 10 groups. (5) **KUST-CBD** includes 10,000 samples from 10 themes with views of Chinese and Burmese documents. (6) **KUST-BTD** includes 10,000 instances with Burmese and Thai documents, across 10 themes. (7) **KUST-CLD** consists of 10,000 instances from 10 themes with Chinese and Laos document views.

The **Reuters** dataset contains 9379 samples from six classes, with English and French views projected into a 10-dimensional space using a standard autoencoder, following the method described by Yang et al. (Yang et al. 2022).

## Comparison Methods And Evaluation Measures

To verify the superiority of our proposed method, we conducted a comprehensive comparison with several state-of-the-art MVC methods, such as **DEMVC** (Xu et al. 2021a); **SDMVC** (Xu et al. 2022a); **DSMVC** (Tang and Liu 2022); **MFLVC** (Xu et al. 2022b); **FastMICE** (Huang, Wang, and Lai 2023); **GCFAgg** (Yan et al. 2023); **DealMVC** (Yang et al. 2023); **CVCL** (Chen et al. 2023); **DIVIDE** (Lu et al. 2024); **ICMVC** (Chao, Jiang, and Chu 2024); and **MAGA** (Bian et al. 2024).

To evaluate clustering performance, we employed four widely accepted metrics: accuracy (ACC), normalized mutual information (NMI), adjusted Rand index (ARI), and purity (PUR). Generally, higher values in these metrics indicate better clustering performance.

## Implementation Details

All samples were reshaped into vectors, and then the fully connected (FC) autoencoders with a similar architecture were used to extract low-dimensional features $h^v$ of multi-view document data. Specifically, the structure of the encoders was given as follows: Input $-$ FC 500 $-$ FC 500 $-$ FC 2000 $-$ FC 512, and the decoders mirrored with the encoder. The following settings were consistent across all experimental datasets: The ReLU activation function was applied to all layers except for the output layer. Adam optimizer was used with a default learning rate of 0.0003. The confidence parameter $\lambda$ was set to 0.9, the focusing factor $\gamma$ to 1, and the temperature parameter $\tau$ to 0.02. For the KUST-CB and KUST-CL datasets, we fixed the parameters $\alpha$ and $\beta$ to 0.001 and 1, respectively. For other datasets, $\alpha$ and $\beta$ were fixed to 0.001 and 0.01. To ensure fairness in comparing results, we ran all the methods five times and reported their average clustering results. The clustering experiments were performed on an Ubuntu computer with an NVIDIA GeForce RTX 3090 GPU (24.0GB memory size).

## Experimental Results And Analysis

In our experiments, we compared our proposed method with several state-of-the-art methods with different metrics. Tables 1 and 2 show the experimental results of various methods on eight datasets. It is worth noting that the best value of the clustering result was highlighted in bold format. It can be seen that our proposed approach outperforms other state-of-the-art methods across various datasets.

On the one hand, our proposed method adopts a multi-level matching strategy to explore the consistency of multi-view data from different perspectives, such as features, pseudo-labels, and prototypes. By minimizing the MMD between the cosine similarity distribution of views and the pseudo-label distribution in the feature space, we can more effectively align the representation of each view. Furthermore, our cross-view prototype matching aligns prototype matrices across views, addressing the prototype offset issue and significantly improving clustering performance.

On the other hand, we introduce an ambiguous instance-aware strategy in contrastive learning. Our instance weight adjustment function dynamically increases the weights of ambiguous instance pairs while reducing the weights of clear instance pairs. It effectively distinguishes positive instance pairs with low similarity and negative instance pairs with high similarity, providing comprehensive training for all instance pairs. As a result, this significantly enhances the network's discriminative ability, leading to better clustering performance.

## Ablation Study

In this subsection, we conducted ablation experiments to assess the contribution of each component in the proposed method under the same experimental settings. Specifically, we constructed six variants of our proposed method: (1) Excluding the similarity distribution matching module, called AICN-MLM (w/o SDM); (2) Removing the cross-view prototype matching module, termed AICN-MLM (w/o CVPM); (3) Eliminating the ambiguous instance aware contrastive learning module, labeled AICN-MLM (w/o AICL); (4) Excluding the ambiguous instance-aware component within the AICL module, referred to as AICN-MLM (w/o AI); (5) Using KL divergence to replace the MMD in the SDM module, referred to as AICN-MLM (w/ KL in SDM); (6) Replacing JS divergence with KL divergence in the CVPM module,

| Method | KUST-ETD | | | | KUST-CTD | | | | KUST-CVD | | | | Reuters | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | ACC | NMI | ARI | PUR | ACC | NMI | ARI | PUR | ACC | NMI | ARI | PUR | ACC | NMI | ARI | PUR |
| DEMVC (2021) | 67.19 | 67.24 | 53.86 | 68.07 | 52.93 | 59.13 | 40.48 | 55.28 | 52.83 | 56.80 | 37.15 | 53.28 | 50.98 | 30.91 | 25.21 | 55.37 |
| SDMVC (2022) | 62.87 | 70.68 | 50.16 | 69.79 | 56.03 | 66.75 | 44.91 | 66.38 | 45.88 | 59.65 | 35.47 | 53.25 | 45.38 | 21.84 | 18.19 | 51.47 |
| DSMVC (2022) | 48.92 | 46.16 | 35.35 | 53.95 | 43.52 | 41.85 | 26.89 | 53.03 | 41.41 | 42.38 | 26.19 | 50.95 | 45.87 | 20.49 | 20.25 | 50.85 |
| MFLVC (2022) | 75.38 | 82.92 | 68.80 | 82.30 | 72.56 | 83.94 | 69.25 | 82.48 | 67.63 | 73.12 | 53.78 | 74.55 | 53.62 | 35.25 | 28.22 | 59.36 |
| FastMICE (2023) | 60.11 | 71.83 | 52.18 | 73.34 | 55.29 | 67.89 | 46.86 | 68.74 | 56.99 | 61.59 | 40.91 | 64.38 | 38.73 | 19.17 | 13.28 | 49.79 |
| GCFAgg (2023) | 74.04 | 81.60 | 66.32 | 80.96 | 73.00 | 81.13 | 65.95 | 79.92 | 62.41 | 74.86 | 52.14 | 74.87 | 55.84 | 37.41 | 29.06 | 59.10 |
| DealMVC (2023) | 74.12 | 78.58 | 66.59 | 74.43 | 74.12 | 78.58 | 66.59 | 74.43 | 71.15 | 76.71 | 57.43 | 78.67 | 55.29 | 42.45 | 31.37 | 62.60 |
| CVCL (2023) | 79.32 | 83.36 | 75.51 | 83.34 | 77.05 | 81.10 | 66.31 | 82.58 | 58.17 | 72.37 | 52.44 | 65.37 | 55.64 | 31.14 | 26.53 | 57.35 |
| DIVIDE (2024) | 55.64 | 70.85 | 41.73 | 79.36 | 70.52 | 78.63 | 62.35 | 77.44 | 58.82 | 65.70 | 52.92 | 69.11 | 50.95 | 34.82 | 27.66 | 59.72 |
| ICMVC (2024) | 69.45 | 78.09 | 61.36 | 78.38 | 56.24 | 64.66 | 42.72 | 66.17 | 63.52 | 74.84 | 56.01 | 72.44 | 51.54 | 36.15 | 28.06 | 59.88 |
| MAGA (2024) | 72.42 | 79.98 | 68.56 | 78.06 | 76.13 | 81.34 | 68.94 | 83.05 | 64.44 | 73.33 | 54.99 | 71.68 | 51.07 | 31.97 | 26.62 | 56.38 |
| AICN-MLM (Ours) | **89.08** | **86.21** | **80.13** | **89.08** | **77.30** | **84.31** | **70.47** | **83.58** | **72.56** | **79.98** | **64.59** | **79.48** | **59.03** | **42.50** | **33.57** | **63.01** |

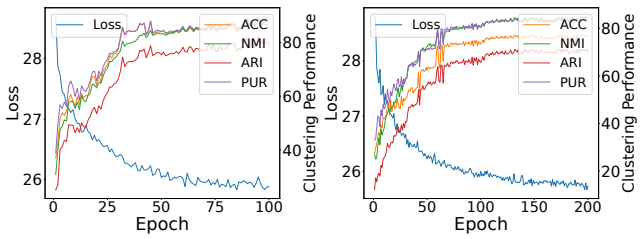
Table 1: The clustering performances of different MVC methods on the KUST-ETD, KUST-CTD, KUST-CVD, and Reuters datasets.

| Method | KUST-CED | | | | KUST-CBD | | | | KUST-BTD | | | | KUST-CLD | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | ACC | NMI | ARI | PUR | ACC | NMI | ARI | PUR | ACC | NMI | ARI | PUR | ACC | NMI | ARI | PUR |
| DEMVC (2021) | 55.67 | 61.25 | 41.76 | 61.43 | 57.22 | 61.49 | 40.75 | 60.53 | 61.39 | 64.39 | 51.93 | 63.08 | 52.23 | 57.56 | 36.54 | 54.88 |
| SDMVC (2022) | 50.44 | 59.81 | 39.19 | 57.96 | 60.46 | 67.75 | 46.85 | 67.35 | 57.60 | 70.42 | 49.20 | 68.43 | 56.08 | 67.47 | 44.85 | 64.66 |
| DSMVC (2022) | 49.02 | 48.11 | 36.24 | 52.80 | 42.94 | 34.89 | 26.85 | 44.96 | 47.75 | 42.05 | 32.08 | 52.56 | 43.03 | 34.58 | 20.15 | 44.24 |
| MFLVC (2022) | 69.75 | 77.80 | 61.84 | 79.14 | 71.35 | 78.86 | 59.10 | 78.27 | 70.83 | 78.61 | 61.61 | 77.75 | 69.14 | 79.99 | 62.63 | 76.85 |
| FastMICE (2023) | 62.81 | 69.34 | 48.93 | 68.57 | 58.43 | 67.05 | 48.62 | 67.72 | 57.96 | 69.92 | 49.73 | 70.83 | 61.58 | 71.65 | 51.51 | 72.18 |
| GCFAgg (2023) | 71.96 | 78.53 | 59.76 | 77.40 | 67.90 | 81.35 | 63.78 | 79.27 | 72.49 | 80.39 | 65.10 | 79.41 | 69.88 | 78.46 | 61.95 | 76.80 |
| DealMVC (2023) | 65.47 | 73.93 | 52.54 | 73.29 | 71.60 | 73.90 | 62.02 | 73.88 | 68.24 | 69.00 | 58.81 | 68.24 | 60.81 | 70.53 | 52.88 | 67.95 |
| CVCL (2023) | 63.66 | 72.97 | 51.43 | 71.91 | 60.22 | 74.37 | 53.88 | 69.30 | 73.54 | 77.87 | 64.10 | 78.87 | 62.81 | 73.58 | 69.72 | 53.23 |
| DIVIDE (2024) | 63.96 | 71.25 | 52.07 | 72.10 | 68.95 | 76.16 | 58.42 | 75.87 | 55.14 | 68.82 | 41.05 | 78.31 | 54.54 | 64.02 | 36.63 | 63.91 |
| ICMVC (2024) | 66.71 | 75.15 | 57.06 | 74.52 | 59.68 | 76.07 | 58.49 | 79.21 | 70.86 | 79.12 | 62.60 | 77.78 | 61.94 | 73.07 | 52.59 | 68.86 |
| MAGA (2024) | 64.98 | 74.12 | 56.83 | 71.38 | 63.66 | 71.61 | 52.90 | 70.40 | 78.67 | 81.98 | 74.55 | 82.47 | 61.23 | 72.51 | 52.72 | 68.25 |
| AICN-MLM (Ours) | **76.22** | **82.64** | **64.54** | **82.67** | **72.73** | **81.36** | **63.85** | **79.65** | **83.14** | **84.68** | **77.18** | **83.53** | **74.21** | **82.40** | **67.17** | **81.13** |

Table 2: The clustering performances of different MVC methods on the KUST-CED, KUST-CBD, KUST-BTD, and KUST-CLD datasets.

| Model Setting | KUST-ETD | | | | KUST-CTD | | | | KUST-CVD | | | | Reuters | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | ACC | NMI | ARI | PUR | ACC | NMI | ARI | PUR | ACC | NMI | ARI | PUR | ACC | NMI | ARI | PUR |
| AICN-MLM (w/o SDM) | 83.76 | 84.84 | 76.68 | 84.18 | 70.25 | 79.36 | 62.35 | 77.33 | 71.15 | 79.82 | 63.92 | 78.07 | 58.94 | 42.44 | 33.42 | 62.92 |
| AICN-MLM (w/o CVPM) | 84.22 | 85.80 | 78.10 | 84.78 | 75.27 | 83.19 | 70.03 | 82.57 | 69.60 | 77.55 | 60.40 | 76.51 | 57.63 | 41.25 | 32.13 | 62.52 |
| AICN-MLM (w/o AICL) | 28.36 | 18.78 | 11.70 | 33.34 | 29.32 | 15.13 | 12.37 | 28.77 | 28.95 | 21.35 | 11.94 | 31.41 | 35.29 | 8.65 | 7.57 | 39.25 |
| AICN-MLM (w/o AI) | 83.15 | 84.75 | 76.30 | 83.49 | 75.78 | 83.62 | 69.86 | 82.77 | 72.24 | 77.68 | 55.67 | 79.16 | 43.15 | 32.68 | 18.45 | 53.26 |
| AICN-MLM (w/ KL in SDM) | 83.09 | 84.45 | 76.30 | 83.61 | 74.78 | 81.59 | 68.25 | 81.33 | 71.16 | 79.78 | 63.13 | 78.08 | 53.12 | 39.52 | 29.35 | 61.11 |
| AICN-MLM (w/ KL in CVPM) | 78.93 | 80.07 | 70.98 | 78.93 | 62.86 | 73.64 | 56.25 | 70.01 | 56.56 | 67.57 | 43.98 | 59.01 | 53.18 | 39.73 | 29.31 | 61.98 |
| AICN-MLM (w/ PSCL) | 58.39 | 72.27 | 41.34 | 70.58 | 69.87 | 78.35 | 62.87 | 77.86 | 61.54 | 69.37 | 32.94 | 68.46 | 47.61 | 38.34 | 24.47 | 57.71 |
| AICN-MLM (Ours) | **89.08** | **86.21** | **80.13** | **89.08** | **77.30** | **84.31** | **70.47** | **83.58** | **72.56** | **79.98** | **64.59** | **79.48** | **59.03** | **42.50** | **33.57** | **63.01** |

Table 3: The ablation study of our method on the KUST-ETD, KUST-CTD, KUST-CVD, and Reuters datasets.




(a) KUST-ETD (b) KUST-CTD

Figure 2: The clustering performances and training loss of our approach on the KUST-ETD and KUST-CTD datasets.

referred to as AICN-MLM (w/ KL in CVPM); (7) Replacing the AICL module with the PSCL contrastive loss, called AICN-MLM (w/ PSCL).

Table 3 shows the ablation results of our proposed method across four different datasets. It can be seen that removing any component from our method or substituting our proposed modules with alternative ones significantly degrades clustering performance. This shows that each component of our proposed method plays a crucial role in enhancing performance in real-world clustering applications.

## Convergence Analysis

In this experiment, we analyzed the convergence rate of the proposed model in clustering. Figure 2 illustrates the clustering performance and training loss of our proposed model on the KUST-ETD and KUST-CTD datasets. It can be seen that the loss of the model steadily decreases and eventually converges to a stable value as training epochs increase. Meanwhile, the clustering performance gradually improves and eventually stabilizes. These convergence results demonstrate the reliability and effectiveness of the proposed method in the MVDC tasks.

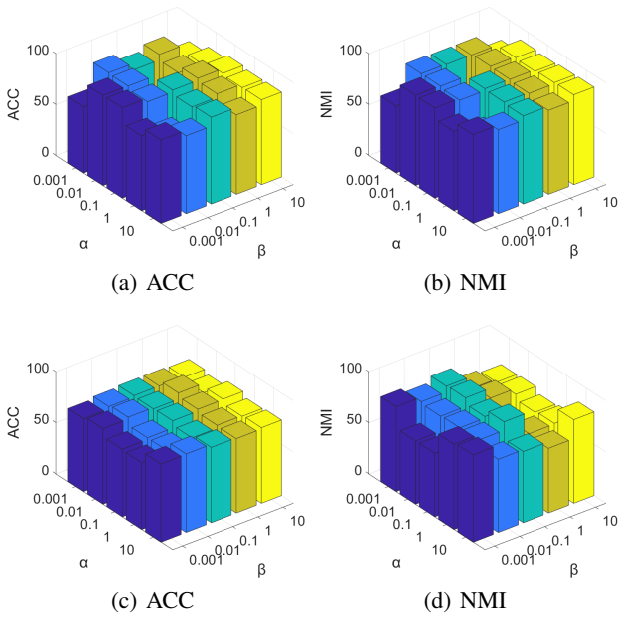
## Parameter Sensitivity Analysis

In this subsection, we comprehensively investigated the impact of hyperparameters $\alpha$ and $\beta$ on the performance of our proposed method. Using a grid search strategy, we tested various combinations of these hyperparameters to analyze their sensitivity and impact on clustering performance.

The range of the hyperparameters $\alpha$ and $\beta$ was set from 0.001 to 10. Figure 3 shows the experimental results of the proposed method on the KUST-ETD dataset with different combinations of hyperparameters. The results show that our method can achieve stable clustering performance across a wide parameter range. This demonstrates that the proposed method can be easily applied to various practical problems.

## Visualization

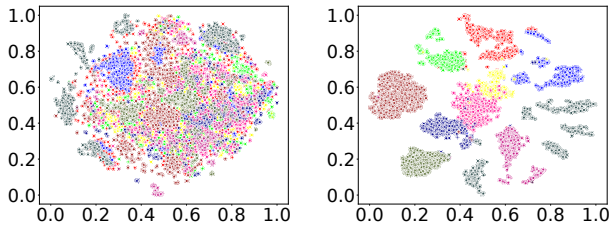
To visually assess the effectiveness of our proposed model, we adopted the $t$-SNE method to display the latent features generated by the clustering layer. Figure 4 shows the visualizations of the raw features and the learned features on the KUST-ETD and KUST-CTD datasets. The results indicate that the clustering structure of the learned features be-




(a) ACC (b) NMI

(c) ACC (d) NMI

Figure 3: The sensitivity analysis of the hyperparameters $\alpha$ and $\beta$ on the KUST-ETD and KUST-CTD datasets.



(a) Raw features (b) Clustering features

Figure 4: The $t$-SNE visualization results of the proposed method on the KUST-ETD dataset.

comes clearer, demonstrating the effectiveness of our proposed model.

## Conclusion

In this paper, we propose a novel method, called ambiguous instance-aware contrastive network with multi-level matching (AICN-MLM), for MvDC tasks. First, we design a multi-level matching strategy from multiple perspectives to more effectively align multi-view features. Additionally, we introduce an ambiguous instance-aware contrastive learning module that adopts an instance weight adjustment function to dynamically increase the weight of ambiguous instances while decreasing the weight of clear instances. This guides the network to focus on ambiguous instances and enhances its discriminative ability, overcoming the limitation of classic contrastive learning that treats all instances equally. Extensive experiments on eight multi-view document datasets demonstrate the superior performance of our proposed method in real-world clustering tasks.

## Acknowledgements

This work was supported by the National Natural Science Foundation of China [Grant 62162033, 6246070439, U21B2027], Yunnan Provincial Major Science and Technology Special Plan Projects [Grant No. 202402AD080001], Yunnan Foundation Research Projects [Grant No. 202101AT070438, 202101BE070001-056], Yunnan Xingdian Talent Support Plan Project.

## References

Bian, J.; Xie, X.; Lai, J.-H.; and Nie, F. 2024. Multi-view contrastive clustering via integrating graph aggregation and confidence enhancement. *Information Fusion*, 108: 102393.

Chao, G.; Jiang, Y.; and Chu, D. 2024. Incomplete Contrastive Multi-View Clustering with High-Confidence Guiding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 11221–11229.

Chao, G.; Sun, S.; and Bi, J. 2021. A survey on multiview clustering. *IEEE transactions on artificial intelligence*, 2(2): 146–168.

Chen, J.; Mao, H.; Woo, W. L.; and Peng, X. 2023. Deep multiview clustering by contrasting cluster assignments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 16752–16761.

Cheng, J.; Wang, Q.; Tao, Z.; Xie, D.; and Gao, Q. 2021. Multi-view attribute graph convolution networks for clustering. In *Proceedings of the twenty-ninth international conference on international joint conferences on artificial intelligence*, 2973–2979.

Chuang, C.-Y.; Robinson, J.; Lin, Y.-C.; Torralba, A.; and Jegelka, S. 2020. Debiased contrastive learning. *Advances in neural information processing systems*, 33: 8765–8775.

Du, G.; Zhou, L.; Yang, Y.; Lü, K.; and Wang, L. 2021. Deep multiple auto-encoder-based multi-view clustering. *Data Science and Engineering*, 6(3): 323–338.

Huang, D.; Wang, C.-D.; and Lai, J.-H. 2023. Fast multi-view clustering via ensembles: Towards scalability, superiority, and simplicity. *IEEE Transactions on Knowledge and Data Engineering*.

Huang, S.; Tsang, I. W.; Xu, Z.; and Lv, J. 2021. Measuring diversity in graph learning: A unified framework for structured multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 34(12): 5869–5883.

Kusner, M. J.; Paige, B.; and Hernández-Lobato, J. M. 2017. Grammar variational autoencoder. In *International conference on machine learning*, 1945–1954. PMLR.

Li, B.; Shu, Z.; Liu, Y.; Mao, C.; Gao, S.; and Yu, Z. 2023. Multi-view clustering via label-embedded regularized NMF with dual-graph constraints. *Neurocomputing*, 551: 126521.

Li, R.; Zhang, C.; Fu, H.; Peng, X.; Zhou, T.; and Hu, Q. 2019a. Reciprocal multi-layer subspace learning for multi-view clustering. In *Proceedings of the IEEE/CVF international conference on computer vision*, 8172–8180.

Li, Z.; Tang, C.; Liu, X.; Zheng, X.; Zhang, W.; and Zhu, E. 2021. Consensus graph learning for multi-view clustering. *IEEE Transactions on Multimedia*, 24: 2461–2472.

Li, Z.; Wang, Q.; Tao, Z.; Gao, Q.; Yang, Z.; et al. 2019b. Deep adversarial multi-view clustering network. In *IJCAI*, volume 2, 4.

Lin, Y.; Gou, Y.; Liu, Z.; Li, B.; Lv, J.; and Peng, X. 2021. Completer: Incomplete multi-view clustering via contrastive prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11174–11183.

Lu, Y.; Lin, Y.; Yang, M.; Peng, D.; Hu, P.; and Peng, X. 2024. Decoupled contrastive multi-view clustering with high-order random walks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 14193–14201.

Shu, Z.; Li, B.; Hu, C.; Yu, Z.; and Wu, X.-J. 2023. Robust dual-graph regularized deep matrix factorization for multi-view clustering. *Neural Processing Letters*, 55(5): 6067–6087.

Shu, Z.; Luo, Y.; Huang, Y.; Mao, C.; and Yu, Z. 2024. View-interactive attention information alignment-guided fusion for incomplete multi-view clustering. *Expert Systems with Applications*, 252: 124258.

Tang, H.; and Liu, Y. 2022. Deep safe multi-view clustering: Reducing the risk of clustering performance degradation caused by view increase. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 202–211.

Wang, Q.; Tao, Z.; Xia, W.; Gao, Q.; Cao, X.; and Jiao, L. 2022a. Adversarial multiview clustering networks with adaptive fusion. *IEEE transactions on neural networks and learning systems*, 34(10): 7635–7647.

Wang, S.; Liu, X.; Liu, S.; Jin, J.; Tu, W.; Zhu, X.; and Zhu, E. 2022b. Align then fusion: Generalized large-scale multi-view clustering with anchor matching correspondences. *Advances in Neural Information Processing Systems*, 35: 5882–5895.

Wang, X.; Zhang, T.; and Gao, X. 2018. Multiview clustering based on non-negative matrix factorization and pairwise measurements. *IEEE transactions on cybernetics*, 49(9): 3333–3346.

Xia, W.; Wang, Q.; Gao, Q.; Zhang, X.; and Gao, X. 2021. Self-supervised graph convolutional network for multi-view clustering. *IEEE Transactions on Multimedia*, 24: 3182–3192.

Xu, J.; Ren, Y.; Li, G.; Pan, L.; Zhu, C.; and Xu, Z. 2021a. Deep embedded multi-view clustering with collaborative training. *Information Sciences*, 573: 279–290.

Xu, J.; Ren, Y.; Tang, H.; Pu, X.; Zhu, X.; Zeng, M.; and He, L. 2021b. Multi-VAE: Learning disentangled view-common and view-peculiar visual representations for multi-view clustering. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9234–9243.

Xu, J.; Ren, Y.; Tang, H.; Yang, Z.; Pan, L.; Yang, Y.; Pu, X.; Philip, S. Y.; and He, L. 2022a. Self-supervised discriminative feature learning for deep multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*.

Xu, J.; Tang, H.; Ren, Y.; Peng, L.; Zhu, X.; and He, L. 2022b. Multi-level feature learning for contrastive multi-view clustering. In *Proceedings of the IEEE/CVF confer-*

*ence on computer vision and pattern recognition*, 16051–16060.

Yan, W.; Zhang, Y.; Lv, C.; Tang, C.; Yue, G.; Liao, L.; and Lin, W. 2023. Gcfagg: Global and cross-view feature aggregation for multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19863–19872.

Yang, M.; Li, Y.; Hu, P.; Bai, J.; Lv, J.; and Peng, X. 2022. Robust multi-view clustering with incomplete information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1): 1055–1069.

Yang, X.; Jiaqi, J.; Wang, S.; Liang, K.; Liu, Y.; Wen, Y.; Liu, S.; Zhou, S.; Liu, X.; and Zhu, E. 2023. Dealmvc: Dual contrastive calibration for multi-view clustering. In *Proceedings of the 31st ACM International Conference on Multimedia*, 337–346.

Yin, Q.; Wu, S.; and Wang, L. 2015. Incomplete multi-view clustering via subspace learning. In *Proceedings of the 24th ACM international on conference on information and knowledge management*, 383–392.

Zhao, H.; Ding, Z.; and Fu, Y. 2017. Multi-view clustering via deep matrix factorization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

Zhao, M.; Yang, W.; and Nie, F. 2023. Deep graph reconstruction for multi-view clustering. *Neural Networks*, 168: 560–568.

Zhou, R.; and Shen, Y.-D. 2020. End-to-end adversarial-attention network for multi-modal clustering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14619–14628.