Investigating and Mitigating the Side Effects of Noisy Views for Self-Supervised Clustering Algorithms in Practical Multi-View Scenarios

Jie Xu[1], Yazhou Ren[1,2], Xiaolong Wang[1], Lei Feng[3], Zheng Zhang[4], Gang Niu[5], Xiaofeng Zhu[1,2,*]

[1]University of Electronic Science and Technology of China, Chengdu, China; [2]Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen, China; [3]Singapore University of Technology and Design, Singapore; [4]Harbin Institute of Technology, Shenzhen, China; [5]RIKEN Center for Advanced Intelligence Project, Tokyo, Japan

Abstract

Multi-view clustering (MVC) aims at exploring category structures among multi-view data in self-supervised manners. Multiple views provide more information than single views and thus existing MVC methods can achieve satisfactory performance. However, their performance might seriously degenerate when the views are noisy in practical multi-view scenarios. In this paper, we formally investigate the drawback of noisy views and then propose a theoretically grounded deep MVC method (namely MVCAN) to address this issue. Specifically, we propose a novel MVC objective that enables un-shared parameters and inconsistent clustering predictions across multiple views to reduce the side effects of noisy views. Furthermore, a two-level multi-view iterative optimization is designed to generate robust learning targets for refining individual views' representation learning. Theoretical analysis reveals that MVCAN works by achieving the multi-view consistency, complementarity, and noise robustness. Finally, experiments on extensive public datasets demonstrate that MVCAN outperforms state-of-the-art methods and is robust against the existence of noisy views.

1. Introduction

Recently, real-world applications generate increasing multi-view data where one sample is described from multiple views, multiple modalities, or multiple groups of features. To handle such multi-view data, multi-view clustering (MVC) is an effective self-supervised clustering approach and has been applied in many fields (*e.g.*, industry [47], internet [12], and medicine [18, 38]), which can recognize the category structures and patterns without label supervision. In addition to traditional MVC [51, 56], deep learning based MVC is usually built on self-supervised methods like contrastive learning [34] and self-training [33], which has been attracting researchers' attention in recent years [6, 20, 39, 42, 62, 71]

and we conduct a review for related work in Appendix A.

The success of existing MVC methods lies in that they are able to explore the *consistency* and *complementarity* among multi-view data [23, 42, 70], thereby outperforming single-view clustering (SVC) methods [27, 30, 37, 57]. The consistency indicates that multiple views have the consistent information which is helpful for recognizing the same category [4, 46, 67]. For example, multiple views with the consistent category information can enhance the recognition of the category semantics, thereby eliminating the interference of non-semantic information. The complementarity means that different views contain the complementary information which is conducive to reciprocally correcting and supplementing each other [22, 41, 60, 64]. In other words, the combination of multiple views can help discover category structures that cannot be discovered by individual views. However, a challenge is that consistency and complementarity of multiple views are still abstract concepts. To conceptually explore them, previous methods usually leverage different views to supervise each other for learning their common representations, and build consistent clustering predictions for all views' agreement. For instance, some methods conduct contrastive learning among multiple views for achieving consistency of representations/predictions [5, 15, 25, 63]. Some methods integrate multiple views' representations to explore their complementarity and generate a unified cluster partition for optimization as self-training manners [49, 55, 58, 62].

Despite important advances, experiments reveal that MVC is not necessarily superior to SVC in some practical multi-view scenarios (see Sec. 4.2). This is because features extracted from some views might be noise, which could be not only useless but even detrimental for clustering. For example, we consider a situation of observing animals at night, where the view captured by infrared cameras is informative but the view from optical cameras is noisy. Contrary to informative views, noisy views can play a negative role in recognizing their common category such that many MVC methods exhibit decreased performance compared to a SVC method that is performed on the optimal single view. This

practical dilemma could affect the effectiveness of MVC and this paper shortly entitles it Noisy-View Drawback (NVD). We find two reasons that the NVD negatively affects the performance of existing MVC methods in practical scenarios: I) To obtain fused representations, many methods have to leverage additional neural networks shared by all views [24, 50, 52, 69, 73]. However, the clustering objective punished on the noisy view might be dominant that on other informative views, causing the shared parameters in that neural networks to fit the noisy view and thus missing the useful information of other views. II) For multi-view data, obtaining consistent clustering predictions for all views is a consensus in previous methods [31, 42, 53, 60, 68, 72]. Nevertheless, it is suboptimal to force the clustering prediction of the noisy view to be the same as that of other views, inversely, this process might make the representation learning and clustering on the informative views degenerate.

In this paper, we consider the NVD and propose a theoretically grounded deep MVC method termed MVCAN: *Multi-View Clustering Against Noisy-view drawback*. Firstly, based on the aforementioned two reasons, the proposed clustering objective I) requires that the parameters in neural networks are un-shared for individual views and II) optimizes a subproblem that allows inconsistent clustering predictions among different views' soft labels. Hence, MVCAN designs parameter-decoupled deep models of learning representations and soft labels for different views, aiming to avoid the side effects of noisy views. Secondly, MVCAN establishes a two-level multi-view iterative optimization for training the parameter-decoupled models. To be exact, $\mathcal{T}$-level leverages the representations and soft labels to optimize a robust learning target, which makes MVCAN able to explore the useful information among informative views and be robust to noisy views. $\mathcal{R}$-level automatically matches the learning target with soft labels to optimize the representations of individual views. Finally, we conduct extensive comparison and ablation experiments to demonstrate the effectiveness of our method. In summary, the contributions of this work include:

- The NVD is pervasive but challenging for MVC, which motivates us to research the robustness towards noisy views. To eliminate the two reasons that noisy views hinder clustering effectiveness, we propose a novel clustering objective constrained with two specific conditions.
- To effectively train a parameter-decoupled model for each view, we propose a two-level multi-view iterative optimization strategy. Extensive experiments on public datasets demonstrate that our MVCAN outperforms state-of-the-art methods and is robust against the existence of noisy views.
- In the literature, almost no work theoretically describes the consistency and complementarity of multi-view learning. This paper attempts to theoretically investigate the consistency and complementarity relations among multiple views, and explain the achieved noise robustness.

2. Background and Analysis

Notations. We denote $\{\mathbf{X}^v \in \mathbb{R}^{N \times D_v}\}_{v=1}^V$ as a multi-view dataset which contains N samples with V views. $\mathbf{Z}^v \in \mathbb{R}^{N \times d_v}$ and $\mathbf{Y}^v \in \mathbb{R}^{N \times K}$ are the learned representations and soft labels for data in the v-th view. D_v and d_v denote the dimensionality of $\mathbf{X}^v$ and $\mathbf{Z}^v$, respectively. K is the cluster number. More notation details are shown in Appendix A.

2.1. Preliminaries

Deep embedded clustering (DEC [57]) is a self-supervised SVC method providing an effective optimization paradigm to promote learning representations and clustering. Specifically, DEC learns representations $\mathbf{Z}$ from data matrix $\mathbf{X}$ of a single view, and conducts end-to-end clustering by learning soft labels $\mathbf{Y}$ with trainable cluster centroids $\{\boldsymbol{\mu}_j\}_{j=1}^K$ in the representation space of $\mathbf{Z}$. DEC formulates $\mathbf{Y}$ as follows:

$$y_{ij} = \frac{(1 + \|\mathbf{z}_i - \boldsymbol{\mu}_j\|_2^2)^{-1}}{\sum_{j=1}^K (1 + \|\mathbf{z}_i - \boldsymbol{\mu}_j\|_2^2)^{-1}} \in \mathbf{Y}, \qquad (1)$$

where $\mathbf{z}_i = \mathcal{E}_{\boldsymbol{\Phi}}(\mathbf{x}_i) \in \mathbf{Z}$ is the new representation of the i-th sample $\mathbf{x}_i \in \mathbf{X}$, obtained by the deep encoder network $\mathcal{E}_{\boldsymbol{\Phi}}$ with the parameters $\boldsymbol{\Phi}$. $(1 + \|\mathbf{z}_i - \boldsymbol{\mu}_j\|_2^2)^{-1}$ can be interpreted as the representation similarity in our Definition 1. We have $\sum_j y_{ij} = 1$ and y_{ij} represents the probabilistic soft label indicating that the sample $\mathbf{x}_i$ comes from the j-th cluster. Then, DEC establishes the learning target $\mathbf{T} \in \mathbb{R}^{N \times K}$ to refine $\mathbf{Y}$ and $\mathbf{Z}$ by training the model parameters, where

$$t_{ij} = \frac{(y_{ij})^2 / \sum_{i=1}^N y_{ij}}{\sum_{j=1}^K \left((y_{ij})^2 / \sum_{i=1}^N y_{ij} \right)} \in \mathbf{T}. \qquad (2)$$

Indeed, Eq. (2) enhances the elements of large values in the soft labels $\mathbf{Y}$ for each sample. As a result, this self-training paradigm establishes the learning target $\mathbf{T}$ to push the soft labels $\mathbf{Y}$ to learn the cluster structures with high confidence.

2.2. Analysis of Noisy-View Drawback (NVD)

The aforementioned learning paradigm inspires a lot of developments and is one of the most widely used approaches to conduct deep MVC [14, 49, 55, 58, 60]. For MVC, previous methods usually learn the representations $\mathbf{Z}^v$ and soft labels $\mathbf{Y}^v$ for individual views, and then leverage the fusion strategies to explore useful information hidden in multiple views, *e.g.*, early fusion [49, 55] and late fusion [58]. They also construct the learning target $\mathbf{T}$ with Eq. (2) to train models.

Although some efforts [45, 49, 55, 66] consider the view diversity and propose weighting strategies in fusion modules, previous methods usually require shared network parameters and consistent clustering predictions for multiple views, whose models might be not robust when meeting low-quality even noisy views in practical scenarios (will be verified in

Sec. 4.2). To illustrate this, we denote $\{\mathbf{Z}^v\}_{v=1}^V$ as all views' representations and consider an ideal clustering objective:

$$\min_{\mathbf{\Theta}} \sum_{v=1}^V \|\mathbf{T} - \mathcal{F}_{\mathbf{\Theta}}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2, \qquad (3)$$

where we write $\mathbf{Y}^v = \mathcal{F}_{\mathbf{\Theta}}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)$ through the fusion module $\mathcal{F}$, and $\mathbf{\Theta}$ denotes the set of parameters shared by all V views. $\mathbf{T}$ is the unified learning target for training the consistent soft labels $\{\mathbf{Y}^v\}_{v=1}^V$ of all views. With the ground-truth label matrix $\mathbf{L} \in \{0,1\}^{N \times K}$, we further have the following theorem to indicate the relationship between clustering effectiveness and clustering objectives of views:

Theorem 1. *Denoting $\check{\mathbf{Y}} = \mathbf{LA}$, where $\mathbf{A} \in \{0,1\}^{K \times K}$ makes $\check{\mathbf{Y}}$ maximally match the learning target $\mathbf{T}$. Then, the clustering accuracy can be calculated as $ACC = \frac{1}{N}\left(N - \frac{1}{2}\|\check{\mathbf{Y}} - \mathbf{T}\|_F^2\right) = 1 - \frac{1}{2N}\|\check{\mathbf{Y}} - \mathbf{T}\|_F^2$. In Eq. (3), if $\mathbf{\Theta}$ is shared by multiple views and their soft labels $\{\mathbf{Y}^v\}_{v=1}^V$ have consistent learning target $\mathbf{T}$, we have*

$$ACC \le 1 - \frac{1}{2N}\bigg(\max_{1 \le m \le V}\|\check{\mathbf{Y}} - \mathcal{F}_{\mathbf{\Theta}}(\mathbf{Y}^m|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2$$
$$- \|\mathbf{T} - \mathcal{F}_{\mathbf{\Theta}}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2\bigg). \qquad (4)$$

To be specific, we denote $m^* = \arg\max_{1 \le m \le V} \|\check{\mathbf{Y}} - \mathcal{F}_{\mathbf{\Theta}}(\mathbf{Y}^m|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2$, and $\|\check{\mathbf{Y}} - \mathcal{F}_{\mathbf{\Theta}}(\mathbf{Y}^{m^*}|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2$ could reflect the largest clustering loss $\|\mathbf{T} - \mathcal{F}_{\mathbf{\Theta}}(\mathbf{Y}^{m^*}|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2$, which corresponds to the view with the worst quality or the most noisy view. No matter how the set of parameters $\mathbf{\Theta}$ is optimized, for the m^*-th view, the unclear cluster structures and inherent noise properties of $\mathbf{Z}^{m^*}$ make it difficult for $\mathbf{Y}^{m^*}$ to fit the learning target $\mathbf{T}$. Therefore, the noisy view has a large clustering loss that is difficult to minimize, *i.e.*, $\|\mathbf{T} - \mathcal{F}_{\mathbf{\Theta}}(\mathbf{Y}^{m^*}|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2$, which usually dominates the optimization of other views in Eq. (3). This makes the shared parameters $\mathbf{\Theta}$ tend to fit the noisy view, resulting the model degeneration on other views which have the small clustering losses that are easy to minimize, *e.g.*, $\sum_{v \ne m^*}\|\mathbf{T} - \mathcal{F}_{\mathbf{\Theta}}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2$. As a consequence, the noisy view will limit the clustering effectiveness due to the upper bound in Eq. (4). The detailed proof and example analysis are provided in Appendix B.

3. Methodology

To mitigate the side effects of noisy views, we propose Multi-View Clustering Against Noisy-View Drawback (MVCAN), whose frame diagram is given in Appendix A due to space.

3.1. Clustering Objective Against NVD

Based on Theorem 1, we consider two conditions to constrain the multi-view clustering objective for MVCAN. The first condition is that we require the network parameters to be decoupled for all views instead of using shared modules when generating $\{\mathbf{Z}^v, \mathbf{Y}^v\}_{v=1}^V$, and the second condition is that we allow different views to have different clustering predictions instead of consistent ones during training stages.

Accordingly, we modify the optimization objective in Eq. (3) and propose a novel multi-view clustering objective:

$$\min_{\{\mathbf{\Theta}^v\}_{v=1}^V} \min_{\{\mathbf{A}^v\}_{v=1}^V} \sum_{v=1}^V \|\mathbf{TA}^v - \mathcal{F}_{\mathbf{\Theta}^v}^v(\mathbf{Y}^v|\mathbf{Z}^v)\|_F^2$$
$$s.t.\ \mathbf{\Theta}^a \cap \mathbf{\Theta}^b = \varnothing, a, b \in \{1, 2, \dots, V\}, a \ne b, \qquad (5)$$
$$\mathbf{A}^v(\mathbf{A}^v)^T = \mathbf{I}_K, \mathbf{A}^v \in \{0,1\}^{K \times K},$$

where we write $\mathbf{Y}^v = \mathcal{F}_{\mathbf{\Theta}^v}^v(\mathbf{Y}^v|\mathbf{Z}^v)$ whose calculation follows Eq. (1). $\mathbf{Z}^v = \mathcal{E}_{\mathbf{\Phi}^v}^v(\mathbf{X}^v)$ and $\mathcal{E}_{\mathbf{\Phi}^v}^v$ denotes the encoder of individual view. Moreover, we specifically illustrate the two conditions in Eq. (5) as follows (their effectiveness will be verified by ablation experiments presented in Sec. 4.3):

Condition 1: In this framework, the set of parameters $\mathbf{\Theta}^v$ includes $\{\mathbf{\mu}_j^v\}_{j=1}^K$ and $\mathbf{\Phi}^v$ of the v-th view. We leverage $\mathbf{\Theta}^a \cap \mathbf{\Theta}^b = \varnothing, a, b \in \{1, 2, \dots, V\}, a \ne b$ to indicate that $\{\mathbf{\Theta}^v\}_{v=1}^V$ are un-shared for each other, so as to avoid the limitations caused by the NVD as analyzed in Sec. 2.2. This condition designs the parameter-decoupled models of learning representations $\{\mathbf{Z}^v\}_{v=1}^V$ and clustering predictions $\{\mathbf{Y}^v\}_{v=1}^V$ for individual views, aiming to eliminate the dominated influence of noisy views on other informative views.

Condition 2: Before updating the parameters $\{\mathbf{\Theta}^v\}_{v=1}^V$, we solve a subproblem in the multi-view clustering objective, that is, $\min_{\{\mathbf{A}^v\}_{v=1}^V} \sum_{v=1}^V \|\mathbf{TA}^v - \mathcal{F}_{\mathbf{\Theta}^v}^v(\mathbf{Y}^v|\mathbf{Z}^v)\|_F^2$, which leads to $\sum_{v=1}^V \|\mathbf{TA}^v - \mathcal{F}_{\mathbf{\Theta}^v}^v(\mathbf{Y}^v|\mathbf{Z}^v)\|_F^2 \le \sum_{v=1}^V \|\mathbf{T} - \mathcal{F}_{\mathbf{\Theta}^v}^v(\mathbf{Y}^v|\mathbf{Z}^v)\|_F^2$. For each view, this subproblem is equivalent to $\min_{\mathbf{A}^v} \|\mathbf{TA}^v - \mathbf{Y}^v\|_F^2$ in which $\mathbf{A}^v \in \{0,1\}^{K \times K}$ achieves the maximum match between the learning target $\mathbf{T}$ and the soft labels $\mathbf{Y}^v$. For each view, $\mathbf{T}$ is adjusted to correspond with $\mathbf{Y}^v$ by $\mathbf{A}^v$ and we can treat this process as to obtain a different learning target $\mathbf{T}^v = \mathbf{TA}^v$. This condition makes the clustering loss smaller, as well as considers that it does not make sense to learn consistent clustering predictions for both informative and noisy views.

3.2. Two-Level Multi-View Iterative Optimization

In brief, to overcome the NVD, MVCAN does not adopt previous strategies that multiple views need shared network parameters and consistent clustering predictions, but how can Eq. (5) explore the useful consistent and complementary information from multiple views? To this end, we propose a two-level multi-view iterative optimization framework for effectively training the parameter-decoupled models.

$\mathcal{T}$**-level iteration**. Firstly, we propose a $\mathcal{T}$-level iteration to generate the robust learning target $\mathbf{T}$ for Eq. (5). $\mathcal{T}$-level iteration will not change the parameters $\{\mathbf{\Theta}^v, \mathbf{A}^v\}_{v=1}^V$, making the models still satisfy the parameter decoupling.

For each iteration in the $\mathcal{T}$-level, we design the scaling matrix $\mathbf{W}_{(t)}$ to automatically explore the informative levels of views for obtaining the scaled representation $\mathbf{Z}_{(t)}$, and then produce the robust soft labels $\mathbf{Y}_{(t)}$ for obtaining $\mathbf{T}$ (the theoretical analysis in Sec. 3.3 will demonstrate that $\mathbf{Y}_{(t)}$ achieve the consistency and complementarity across multiple views, as well as the noise robustness for the noisy views).

Concretely, in the t-th iteration of $\mathcal{T}$-level, MVCAN infers the scaled representations $\mathbf{Z}_{(t)} \in \mathbb{R}^{N \times \sum_v d_v}$ from all views, by the multiplication between the already scaled/normalized representations $\begin{bmatrix} \mathbf{Z}^1 & \mathbf{Z}^2 & \dots & \mathbf{Z}^V \end{bmatrix} \in \mathbb{R}^{N \times \sum_v d_v}$ and the scaling matrix $\mathbf{W}_{(t)} \in \mathbb{R}^{\sum_v d_v \times \sum_v d_v}$:

$$
\begin{aligned}
\mathbf{Z}_{(t)} &= \mathcal{H}(\mathbf{Z}_{(t)}|\mathbf{W}_{(t)}, \mathbf{Z}^1, \mathbf{Z}^1, \dots, \mathbf{Z}^V) \\
&= \begin{bmatrix} \mathbf{Z}^1 & \mathbf{Z}^2 & \dots & \mathbf{Z}^V \end{bmatrix} \mathbf{W}_{(t)} \\
&= \begin{bmatrix} \mathbf{Z}^1 & \mathbf{Z}^2 & \dots & \mathbf{Z}^V \end{bmatrix} \begin{bmatrix} w_{(t)}^1 \mathbf{I}^1 & & & \\ & w_{(t)}^2 \mathbf{I}^2 & & \\ & & \ddots & \\ & & & w_{(t)}^V \mathbf{I}^V \end{bmatrix},
\end{aligned}
$$
(6)

where $\mathbf{W}_{(t)}$ is a block diagonal matrix ($\mathbf{W}_{(1)} = \mathbf{I}$), of which each block is the multiplication between the unit matrix $\mathbf{I}^v \in \{0,1\}^{d_v \times d_v}$ and the scaling factor $w_{(t)}^v \in \mathbb{R}$ for the individual view. Based on the scaled representations $\mathbf{Z}_{(t)}$, MVCAN generates the robust soft labels $\mathbf{Y}_{(t)} \in \mathbb{R}^{N \times K}$ in the t-th iteration. To be specific, $\mathbf{Y}_{(t)}$ should reflect the cluster structures among $\mathbf{Z}_{(t)}$, and thus we leverage a variant of Eq. (1) to compute $\mathbf{Y}_{(t)}$. Specifically, $\mathbf{z}_{i(t)} \in \mathbf{Z}_{(t)}$ and we formulate $\mathbf{Y}_{(t)} = \mathcal{F}'(\mathbf{Y}_{(t)}|\mathbf{Z}_{(t)})$ as follows:

$$
y_{ij(t)} = \frac{(1 + \|\mathbf{z}_{i(t)} - \mathbf{c}_{j(t)}\|_2^2)^{-1}}{\sum_{j=1}^K (1 + \|\mathbf{z}_{i(t)} - \mathbf{c}_{j(t)}\|_2^2)^{-1}} \in \mathbf{Y}_{(t)}, \quad (7)
$$

where $\{\mathbf{c}_{j(t)} \in \mathbb{R}^{\sum_v d_v}\}_{j=1}^K$ represent the cluster centroids of $\mathbf{Z}_{(t)}$ in the t-th iteration. Note that $\{\mathbf{c}_{j(t)}\}_{j=1}^K$ are computed by K-means [27] from the scratch in each iteration, it will not change the parameters $\{\mathbf{\Theta}^v, \mathbf{A}^v\}_{v=1}^V$. Furthermore, denoting I and H as mutual information and entropy, respectively, we base on the normalized mutual information between the robust soft labels $\mathbf{Y}_{(t)}$ and the soft labels $\mathbf{Y}^v$ of individual view, and denote the iterative strategy of the scaling matrix as $\mathbf{W}_{(t+1)} = \mathcal{G}(\mathbf{W}_{(t+1)}|\mathbf{Y}_{(t)}, \mathbf{Y}^1, \mathbf{Y}^1, \dots, \mathbf{Y}^V)$, in which we compute $w_{(t+1)}^v$ for each view by

$$
w_{(t+1)}^v = \exp\left(\frac{2I(\mathbf{Y}^v; \mathbf{Y}_{(t)})}{H(\mathbf{Y}^v) + H(\mathbf{Y}_{(t)})} \right) \in \mathbf{W}_{(t+1)}. \quad (8)
$$

To effectively calculate Eq. (8), we first transform $\mathbf{Y}^v$ and $\mathbf{Y}_{(t)}$ into one-dimensional label vectors $\hat{\mathbf{y}}^v$ and $\hat{\mathbf{y}}_{(t)}$, respectively, where $\hat{y}_i^v = \arg\max_j y_{ij}^v$, $\hat{y}_{i(t)} = \arg\max_j y_{ij(t)}$, and then calculate the normalized mutual information between $\hat{\mathbf{y}}^v$ and $\hat{\mathbf{y}}_{(t)}$. Since the computations of $\mathbf{Y}_{(t)}$ and $\{\mathbf{Y}^v\}_{v=1}^V$ are all un-/self-supervised, in effect, MVCAN can

automatically recognize the informative levels of different views based on the mutual information among the soft labels, and then generate different scaling factors in $\mathbf{W}_{(t+1)}$ to constrain the representations of all views for next iterations.

After finishing the iteration of the robust soft labels $\mathbf{Y}_{(t)}$, we utilize Eq. (2) to obtain the robust learning target, written as $\mathbf{T} = \mathcal{T}(\mathbf{T}|\mathbf{Y}_{(t)})$. Hence, the robust learning target $\mathbf{T}$ is based on the already learned representations and soft labels, i.e., $\{\mathbf{Z}^v, \mathbf{Y}^v\}_{v=1}^V$. The $\mathcal{T}$-level iteration process outputs $\mathbf{T}$ which is further leveraged to refine $\{\mathbf{Z}^v, \mathbf{Y}^v\}_{v=1}^V$ for all views by the multi-view clustering objective in Eq. (5).

$\mathcal{R}$**-level iteration**. $\mathcal{R}$-level iteration focuses on training the parameters $\{\mathbf{\Theta}^v, \mathbf{A}^v\}_{v=1}^V$ for individual views by optimizing Eq. (5). Considering the Condition 2 of Eq. (5), we first obtain $\mathbf{A}^{v*} = \min_{\mathbf{A}^v} \|\mathbf{TA}^v - \mathbf{Y}^v\|_F^2$ with Hungarian algorithm. For each view, $\mathbf{A}^{v*}$ produces a different learning target $\mathbf{T}^v = \mathbf{TA}^{v*}$ and then Eq. (5) can be transformed into the following clustering objective (denoted by $\mathcal{L}_c^v$):

$$
\mathcal{L}_c^v : \min_{\mathbf{\Theta}^v} \|\mathbf{T}^v - \mathcal{F}_{\mathbf{\Theta}^v}^v(\mathbf{Y}^v|\mathbf{Z}^v)\|_F^2. \quad (9)
$$

Additionally, we follow previous deep MVC methods [43, 49, 55, 58, 62] and adopt deep autoencoders (a popular self-supervised representation learning method) to learn the new representations of multi-view data. Letting $\mathcal{E}_{\mathbf{\Phi}^v}^v$ and $\mathcal{D}_{\mathbf{\Psi}^v}^v$ respectively denote the encoder and decoder, our method requires that the network parameters $\mathbf{\Phi}^v$ and $\mathbf{\Psi}^v$ of each view are un-shared for other views according to the Condition 1 of Eq. (5). Therefore, for the v-th view, the reconstruction $\hat{\mathbf{X}}^v = \mathcal{D}_{\mathbf{\Psi}^v}^v(\mathbf{Z}^v)$ is only related to $\mathbf{Z}^v = \mathcal{E}_{\mathbf{\Phi}^v}^v(\mathbf{X}^v)$, and the representation learning objective (denoted by $\mathcal{L}_r^v$) is:

$$
\mathcal{L}_r^v : \min_{\{\mathbf{\Psi}^v, \mathbf{\Phi}^v\}} \|\mathbf{X}^v - \mathcal{D}_{\mathbf{\Psi}^v}^v(\mathcal{E}_{\mathbf{\Phi}^v}^v(\mathbf{X}^v))\|_F^2. \quad (10)
$$

In $\mathcal{R}$-level iteration, the loss function to train the parameter-decoupled model of each view includes following two parts:

$$
\mathcal{L}^v = \mathcal{L}_r^v + \lambda \mathcal{L}_c^v, \quad (11)
$$

where λ achieves the trade-off between $\mathcal{L}_r^v$ and $\mathcal{L}_c^v$. Meanwhile, we have $\{\mathbf{\Theta}^a, \mathbf{\Psi}^a, \mathbf{\Phi}^a\} \cap \{\mathbf{\Theta}^b, \mathbf{\Psi}^b, \mathbf{\Phi}^b\} = \varnothing, a, b \in \{1, 2, \dots, V\}, a \neq b$ which overcomes the mutual interference among different views during training their network parameters. The $\mathcal{R}$-level iteration process refines the representations and soft labels $\{\mathbf{Z}^v, \mathbf{Y}^v\}_{v=1}^V$ which are further leveraged to obtain better learning target $\mathbf{T}$. At last, $\mathbf{Y}_{(t)}$ outputs clustering results for all multi-view data and Algorithm 1 concludes the training steps of MVCAN (the effectiveness of two losses and iterations will be verified in Sec. 4.3).

3.3. Theoretical Analysis of Multi-View Consistency & Complementarity & Noise Robustness

Moreover, we attempt to theoretically illustrate why MV-CAN works with the following definitions and theorems:

4

Algorithm 1: Training steps of MVCAN

Input: Dataset $\{\mathbf{X}^v\}_{v=1}^V$, Epochs E, T_1, T_2, K, λ

Initialize $\{\mathbf{\Phi}^v, \mathbf{\Psi}^v\}_{v=1}^V$ by Eq. (10) and initialize
$\{\{\mathbf{\mu}_j^v\}_{j=1}^K\}_{v=1}^V$ with K-means, $\mathbf{W}_{(1)} = \mathbf{I}$

for $e \in \{1, 2, \ldots, E/T_2\}$ **do**

    // $\mathcal{T}$-level infers $\mathbf{T}$ from all views' $\{\mathbf{Z}^v, \mathbf{Y}^v\}_{v=1}^V$.

    for $t \in \{1, 2, \ldots, T_1\}$ **do**

        Update $\mathbf{Z}_{(t)}$ by Eq. (6)

        Update $\mathbf{Y}_{(t)}$ by Eq. (7)

        Update $\mathbf{W}_{(t+1)}$ by Eq. (8)

    Update $\mathbf{T} = \mathcal{T}(\mathbf{T}|\mathbf{Y}_{(t)})$ as Eq. (2)

    // $\mathcal{R}$-level learns $\{\mathbf{Z}^v, \mathbf{Y}^v\}$ for each view with $\mathbf{T}$.

    for $v \in \{1, 2, \ldots, V\}$ **do**

        Compute $\mathbf{A}^v$ by $\min_{\mathbf{A}^v} \|\mathbf{T}\mathbf{A}^v - \mathbf{Y}^v\|_F^2$ in
        Eq. (5) with Hungarian algorithm

        Update $\mathbf{\Phi}^v$, $\mathbf{\Psi}^v$, and $\{\mathbf{\mu}_j^v\}_{j=1}^K$ on $\mathbf{X}^v$ for T_2
        epochs by Eq. (11) with mini-batch Adam

Output: The cluster assignment of the i-th sample
$\arg\max_j y_{ij(t)}$ where $y_{ij(t)} \in \mathbf{Y}_{(t)}$, all views'
model parameters $\{\mathbf{\Phi}^v, \mathbf{\Psi}^v, \{\mathbf{\mu}_j^v\}_{j=1}^K, \mathbf{A}^v\}_{v=1}^V$

Definition 1. *Denoting* $\mathcal{D}(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\|_2^2$ *as squared Euclidean distance between the representations* $\mathbf{a}$ *and* $\mathbf{b}$,

$$\mathcal{S}(\mathbf{a}, \mathbf{b}) := \frac{1}{1 + \mathcal{D}(\mathbf{a}, \mathbf{b})} \in (0, 1] \qquad (12)$$

is defined as the representation similarity between $\mathbf{a}$ *and* $\mathbf{b}$. *Formally,* $y_{ij}^v \in (0, 1]$ *holds given Eq. (1).*

Definition 2. *(ε, $\mathbf{z}$, $\mathbf{\mu}$ - Noisy-view) For* $\forall \mathbf{z}_i^v \in \mathbf{Z}^v$, *it belongs to the noisy view if* $\exists \mathbf{\mu}_a^v, \mathbf{\mu}_b^v$, *and* $\varepsilon > 0$ *such that* $|\mathcal{D}(\mathbf{z}_i^v, \mathbf{\mu}_a^v) - \mathcal{D}(\mathbf{z}_i^v, \mathbf{\mu}_b^v)| < \varepsilon$, $\mathcal{S}(\mathbf{z}_i^v, \mathbf{\mu}_a^v) \approx \mathcal{S}(\mathbf{z}_i^v, \mathbf{\mu}_b^v)$, *and* $y_{ia}^v \approx y_{ib}^v$, *where* ε *is a sufficiently small value. Otherwise,* $\mathbf{z}_i^v$ *is the informative view.*

Then, the following theorems suggest that $\mathbf{Y}_{(t)}$ achieves our concluded consistency, complementarity, and noise robustness with regard to $\{\mathbf{Y}^v\}_{v=1}^V$ in the framework of MVCAN. All proofs of theorems are provided in Appendix B.

Theorem 2. *Denoting* $\mathcal{L}_K$ *as the K-means objective,* $\mathcal{L}_K(\mathbf{Z}_{(t)})$ *is equivalent to punishing different scaling factors on* $\{\mathcal{L}_K(\mathbf{Z}^v)\}_{v=1}^V$ *under the consistency constraint of multiple views' cluster centroids.*

Theorem 2 analyses the effect of the scaling matrix $\mathbf{W}_{(t)}$ to constrain the optimization of individual views in the scaled representation $\mathbf{Z}_{(t)}$, which reduces the side effects of noisy views when $\mathcal{T}$-level iteration discovers the cluster structures.

Theorem 3. *(Consistency) If a sample representation is informative in multiple views and has the same cluster assignments in these views, its cluster assignment in* $\mathbf{Y}_{(t)}$ *is the same as that in these views.*

Theorem 3 indicates that $y_{ij(t)} \in \mathbf{Y}_{(t)}$ follows $\{y_{ij}^v \in \mathbf{Y}^v\}_{v=1}^V$ when they have consistent clusters, which reflects the property of consistency among multiple views.

Theorem 4. *(Complementarity) If a sample representation is informative in multiple views where it has different cluster assignments, we have two cases according to the differences of similarity among the clusters.*
Case 1: if the differences of similarity among the clusters are equal, its cluster assignment in $\mathbf{Y}_{(t)}$ *is the same as that in the informative view with the largest scaling factor.*
Case 2: if the differences of similarity among the clusters are not equal, its cluster assignment in $\mathbf{Y}_{(t)}$ *is more likely to be the same as that in the informative view with the largest scaling factor.*

Theorem 4 indicates that $y_{ij(t)} \in \mathbf{Y}_{(t)}$ follows $y_{ij}^v \in \mathbf{Y}^v$ with a large scaling factor when different views have inconsistent clusters, which leverages the view with high confidence to correct other inconsistent views.

Theorem 5. *(Complementarity & Noise robustness)*
Case 1: if a sample representation is informative in some views and is noisy in other views, its cluster assignment in $\mathbf{Y}_{(t)}$ *is the same as that in the informative views.*
Case 2: if a sample representation is noisy in all views, its cluster assignment in $\mathbf{Y}_{(t)}$ *is the same as the common cluster assignments existing in these views.*

Theorem 5 illustrates the noise robustness of our method that makes the robust soft labels $\mathbf{Y}_{(t)}$ mitigate the side effects of noisy views. For example, $\mathbf{z}_i^v \in \mathbf{Z}^v$ is noisy in individual view but the corresponding scaled representation $\mathbf{z}_{i(t)} \in \mathbf{Z}_{(t)}$ is informative, so the influence from noisy views on the soft labels $y_{ij(t)} \in \mathbf{Y}_{(t)}$ are reduced. Additionally, Theorems 4 and 5 can be together interpreted as the complementarity among multiple views, *i.e.*, the combination of multiple views is conducive to outperforming single views and discovering comprehensive cluster patterns (which cannot be explored in single-view data) across multi-view data.

4. Experiments

4.1. Settings

We briefly introduce the experimental setup and show more implementation details in Appendix. Our code is provided in https://github.com/SubmissionsIn/MVCAN.

Datasets. We conduct experiments on eight public datasets and four noise-simulated ones, and their details are listed in Appendix C. First, four normal multi-view datasets (easy for clustering) include BDGP [3], DIGIT [36], COIL [29], and Amazon [40]. Second, we construct four noise-simulated datasets on the four datasets to test the noise robustness of methods in extreme scenarios, where we randomly sample noise to build an additional view and obtain

Table 1. Clustering performance gains (%) of MVC methods compared with SVC method (DEC-BestV) on four normal multi-view datasets.

Method	BDGP		DIGIT		COIL		Amazon	
	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
DEC-BestV [57]	92.6	81.9	80.9	78.9	76.6	81.5	47.0	32.5
DEC-WorstV [57]	$45.7_{-46.9}$	$29.1_{-52.8}$	$54.8_{-26.1}$	$64.1_{-14.8}$	$73.5_{-3.1}$	$77.4_{-4.1}$	$37.2_{-9.8}$	$27.9_{-4.6}$
DMJC [58]	$67.8_{-24.8}$	$46.5_{-35.4}$	$97.6_{+16.7}$	$96.2_{+17.3}$	$91.3_{+14.7}$	$93.8_{+12.3}$	$63.3_{+16.3}$	$65.3_{+32.8}$
DIMC-net [55]	$97.5_{+4.9}$	$91.1_{+9.2}$	$90.4_{+9.5}$	$87.3_{+8.4}$	$98.5_{+21.9}$	$97.5_{+16.0}$	$62.5_{+15.5}$	$66.9_{+34.4}$
GP-MVC [49]	$97.6_{+5.0}$	$93.4_{+11.5}$	$58.6_{-22.3}$	$69.8_{-9.1}$	$86.1_{+9.5}$	$77.5_{-4.0}$	$53.9_{+6.9}$	$57.1_{+24.6}$
CoMVC [45]	$80.7_{-11.9}$	$67.4_{-14.5}$	$98.5_{+17.6}$	$97.4_{+18.5}$	$98.1_{+21.5}$	$97.8_{+16.3}$	$68.1_{+21.1}$	$60.6_{+28.1}$
DIMVC [60]	$98.1_{+5.5}$	$93.8_{+11.9}$	$97.6_{+16.7}$	$96.0_{+17.1}$	$93.4_{+16.8}$	$93.5_{+12.0}$	$77.1_{+30.1}$	$81.3_{+48.8}$
DSMVC [42]	$52.9_{-39.7}$	$38.3_{-43.6}$	$82.0_{+1.1}$	$81.4_{+2.5}$	$90.8_{+14.2}$	$96.5_{+15.0}$	$37.6_{-9.4}$	$29.2_{-3.3}$
DSIMVC [43]	$98.0_{+5.4}$	$94.0_{+12.1}$	$99.0_{+18.1}$	$97.1_{+18.2}$	$99.7_{+23.1}$	$99.0_{+17.5}$	$64.6_{+17.6}$	$57.8_{+25.3}$
CPSPAN [15]	$91.5_{-1.1}$	$77.2_{-4.7}$	$84.8_{+3.9}$	$82.1_{+3.2}$	$80.4_{+3.8}$	$85.1_{+3.6}$	$71.2_{+24.2}$	$60.8_{+28.3}$
SDMVC [62]	$98.5_{+5.9}$	$95.0_{+13.1}$	$99.8_{+18.9}$	$99.5_{+20.6}$	$97.0_{+20.4}$	$95.6_{+14.1}$	$57.9_{+10.9}$	$66.5_{+34.0}$
MVCAN [ours]	$98.4_{+5.8}$	$95.3_{+13.4}$	$99.5_{+18.6}$	$98.8_{+19.9}$	$99.6_{+23.0}$	$99.1_{+17.6}$	$82.6_{+35.6}$	$86.7_{+54.2}$

Table 2. Clustering performance gains (%) of MVC methods compared with SVC method (DEC-BestV) on four real-world multi-view datasets. "n/a" denotes the unavailable clustering result due to high computational costs.

Method	DHA		RGB-D		Caltech		YoutubeVideo	
	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
DEC-BestV [57]	72.6	79.3	43.6	40.1	88.2	81.6	20.9	20.4
DEC-WorstV [57]	$30.4_{-42.2}$	$43.5_{-35.8}$	$15.0_{-28.6}$	$5.1_{-35.0}$	$35.4_{-52.8}$	$19.6_{-62.0}$	$26.6_{+5.7}$	$0.0_{-20.4}$
DMJC [58]	$64.4_{-8.2}$	$73.9_{-5.4}$	$31.7_{-11.9}$	$28.5_{-11.6}$	$83.1_{-5.1}$	$80.3_{-1.3}$	$15.1_{-5.8}$	$15.3_{-5.1}$
DIMC-net [55]	$60.3_{-12.3}$	$73.5_{-5.8}$	$35.6_{-8.0}$	$32.4_{-7.7}$	$75.0_{-13.2}$	$68.5_{-13.1}$	n/a	n/a
GP-MVC [49]	$73.1_{+0.5}$	$81.5_{+2.2}$	$38.5_{-5.1}$	$32.6_{-7.5}$	$80.3_{-7.9}$	$77.6_{-4.0}$	$12.4_{-8.5}$	$10.3_{-10.1}$
CoMVC [45]	$67.4_{-5.2}$	$79.2_{-0.1}$	$42.0_{-1.6}$	$41.3_{+1.2}$	$72.5_{-15.7}$	$68.8_{-12.8}$	$18.1_{-2.8}$	$17.9_{-2.5}$
DIMVC [60]	$79.5_{+6.9}$	$84.7_{+5.4}$	$46.9_{+3.3}$	$41.4_{+1.3}$	$87.2_{-1.0}$	$80.7_{-0.9}$	$15.4_{-5.5}$	$12.5_{-7.9}$
DSMVC [42]	$77.4_{+4.8}$	$83.6_{+4.3}$	$43.3_{-0.3}$	$40.6_{+0.5}$	$90.5_{+2.3}$	$84.7_{+3.1}$	$17.8_{-3.1}$	$18.0_{-2.4}$
DSIMVC [43]	$64.0_{-8.6}$	$77.3_{-2.0}$	$45.8_{+2.2}$	$41.0_{+0.9}$	$76.7_{-11.5}$	$67.5_{-14.1}$	$19.0_{-1.9}$	$18.8_{-1.6}$
CPSPAN [15]	$67.1_{-5.5}$	$80.0_{+0.7}$	$42.4_{-1.2}$	$38.3_{-1.8}$	$84.8_{-3.4}$	$73.9_{-7.7}$	$23.0_{+2.1}$	$22.0_{+1.6}$
SDMVC [62]	$80.2_{+7.6}$	$85.4_{+6.1}$	$44.1_{+0.5}$	$40.7_{+0.6}$	$85.3_{-2.9}$	$79.1_{-2.5}$	$18.6_{-2.3}$	$18.0_{-2.4}$
MVCAN [ours]	$84.8_{+12.2}$	$87.5_{+8.2}$	$48.0_{+4.4}$	$41.7_{+1.6}$	$93.6_{+5.4}$	$88.7_{+7.1}$	$24.2_{+3.3}$	$24.3_{+3.9}$

NoisyBDGP/DIGIT/COIL/Amazon for the four individual datasets. Third, we conduct experiments on four real-world multi-view datasets (hard for clustering) including DHA [21], RGB-D [72], Caltech [8], and YoutubeVideo [28].

Comparison methods. We compare our MVCAN with the following 10 self-supervised clustering algorithms. To be specific, DEC [57] is a popular deep SVC method and we leverage this baseline to investigate the side effects of NVD on MVC methods. DMJC [58], DIMC-net [55], GP-MVC [49], DIMVC [60], and SDMVC [62] are DEC-based deep MVC methods which usually establish consistent soft labels for achieving clustering consistency. DMJC [58], DIMC-net [55], GP-MVC [49], and DSMVC [42] mainly incorporate weighting strategies to obtain fused representations. CoMVC [45], DSIMVC [43], and CPSPAN [15] are contrastive learning based deep MVC methods which leverage contrastive learning to learn common representations.

4.2. Comparison Results and Analysis

Tables 1, 2, and 3 list clustering effectiveness of comparison methods on all datasets. The performance is evaluated by clustering accuracy (ACC) and normalized mutual information (NMI), and the average values of 10 runs are reported.

DEC-BestV and DEC-WorstV denote the results of the SVC method DEC on the best and the worst views, respectively.

Firstly, we compare DEC-BestV with DEC-WorstV and can easily find that the clustering results of DEC-WorstV is not ideal for many samples, that is, many samples that are correctly clustered by DEC-BestV are incorrectly clustered by DEC-WorstV (especially for real-world multi-view datasets in Table 2). This suggests that the view qualities of multi-view datasets are different, where the views with unclear cluster structures could be considered as noisy views for clustering. Secondly, most of MVC methods achieve performance gains on normal datasets (red results in Table 1) but have performance degeneration on real-world datasets (green results in Table 2) when taking DEC-BestV as the baseline. The side effects of noisy views adversely affect many MVC methods and thus we observe that some multi-view methods are not robuster than the single-view method in terms of clustering effectiveness. Despite some of these MVC methods leverage weighting strategies to balance different views, the noisy-view drawback still prevent them from learning effective cluster structures in some practical scenarios. Thirdly, our method MVCAN obtains much better performance than DEC-BestV across all datasets and generally achieves the best or comparable performance among all

Table 3. Clustering performance gains (%) of MVC methods compared with SVC method (DEC-BestV) on four noise-simulated datasets.

Method	NoisyBDGP		NoisyDIGIT		NoisyCOIL		NoisyAmazon	
	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
DEC-BestV [57]	92.6	81.9	80.9	78.9	76.6	81.5	47.0	32.5
DEC-WorstV [57]	22.2-70.4	0.2-81.7	12.4-68.5	0.4-78.5	16.4-60.2	2.8-78.7	12.0-35.0	0.4-32.1
DMJC [58]	63.7-28.9	59.4-22.5	80.7-0.2	82.8+3.9	85.6+9.0	92.1+10.6	54.3+7.3	46.6+14.1
DIMC-net [55]	78.9-13.7	68.4-13.5	71.6-9.3	76.5-2.4	87.5+10.9	91.8+10.3	43.6-3.4	37.3+4.8
GP-MVC [49]	80.7-11.9	78.4-3.5	49.1-31.8	63.5-15.4	69.4-7.2	72.9-8.6	40.4-6.6	39.8+7.3
CoMVC [45]	63.8-28.8	51.6-30.3	86.9+6.0	84.6+5.7	90.6+14.0	93.6+12.1	61.8+14.8	52.6+20.1
DIMVC [60]	94.9+2.3	87.6+5.7	88.7+7.8	93.7+14.8	89.0+12.4	91.7+10.2	63.6+16.6	66.7+34.2
DSMVC [42]	57.1-35.5	41.8-40.1	73.7-7.2	72.2-6.7	81.8+5.2	84.1+2.6	36.6-10.4	25.9-6.6
DSIMVC [43]	95.1+2.5	85.2+3.3	90.4+9.5	90.5+11.6	98.8+22.2	97.8+16.3	54.7+7.7	54.1+21.6
CPSPAN [15]	73.2-19.4	53.5-28.4	11.8-69.1	0.3-78.6	15.8-60.8	3.3-78.2	12.4-34.6	0.4-32.1
SDMVC [62]	89.6-3.0	83.6+1.7	75.8-5.1	72.2-6.7	81.0+4.4	89.2+7.7	55.4+8.4	61.0+28.5
MVCAN [ours]	98.0+5.4	95.1+13.2	99.0+18.1	98.4+19.5	99.2+22.6	98.8+17.3	72.8+25.8	73.2+40.7

MVC methods. For example in Table 2, MVCAN improves the best comparison methods by 4%, 1%, and 3% ACC values on DHA, RGB-D, and Caltech, respectively. The results indicate that MVCAN is able to explore the useful consistent and complementary information among informative views, as well as achieve the noise robustness to noisy views.

Since the performance of MVC could be interfered with noisy views in datasets, it is encouraged to test the robustness of algorithms with extreme noise interference [45, 66], which can guide the algorithm design of MVC for practical scenarios. To this end, we conduct comparison experiments on noise-simulated multi-view datasets as shown in Table 3. Compared with Table 1, Table 3 suggests that most of MVC methods have degenerated results but our MVCAN still achieves comparable performance. Specifically, MVCAN surpasses the best comparison methods by 7%, 5%, 1%, and 6% NMI values on the four noise-simulated datasets. This further demonstrates the effectiveness of our method.

4.3. Ablation Study

In this subsection, we investigate the effectiveness of each part of our method in detail from the following aspects.

Table 4. Importance of two conditions in clustering objective.

	Conditions		BDGP		DIGIT		NoisyBDGP		NoisyDIGIT	
	Θ^v	$\mathbf{A}^v$	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
(i)	✓		97.3	92.1	98.6	98.0	97.7	94.6	90.2	93.3
(ii)		✓	66.0	47.7	84.0	73.5	60.2	33.9	61.1	59.9
(iii)	✓	✓	98.4	95.3	99.5	98.8	98.0	95.1	99.0	98.4

Table 5. Importance of two loss components in optimization.

	Components		BDGP		DIGIT		NoisyBDGP		NoisyDIGIT	
	$\mathcal{L}_r^v$	$\mathcal{L}_c^v$	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
(a)			64.3	52.2	76.8	72.3	49.9	31.5	74.1	70.5
(b)	✓		94.8	84.2	78.7	74.7	94.4	83.9	76.9	74.8
(c)		✓	79.8	70.9	87.0	94.3	72.6	57.4	59.1	70.2
(d)	✓	✓	98.4	95.3	99.5	98.8	98.0	95.1	99.0	98.4

Two conditions in clustering objective. We investigate the importance of the two conditions in Eq. (5). As shown
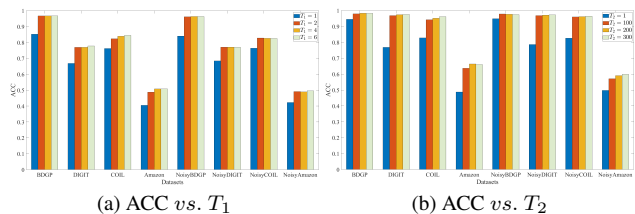

Figure 1. Different training iterations of $\mathcal{T}$-level (a) and $\mathcal{R}$-level (b) in the proposed two-level multi-view iterative optimization.

in Table 4, (i) $\boldsymbol{\Theta}^v$ denotes the first condition of un-shared parameters for all views and (ii) $\mathbf{A}^v$ indicates the second condition that multiple views are not required to be consistent. One could find that the results shown in (iii) achieve the best performance, which verifies the effectiveness of our MVCAN to mitigate the side effects caused by the noisy-view drawback. Concretely, the un-shared $\{\boldsymbol{\Theta}^v\}_{v=1}^V$ of all views eliminate their unfavourable interference. Moreover, $\{\mathbf{A}^v\}_{v=1}^V$ absolve the noisy views of conformity with the other views when minimizing the clustering objective.

Two loss components in optimization. Table 5 lists the results of MVCAN with different loss components, where (a) denotes the clustering results of K-means on the direct concatenation of multi-view data. Compared with (a), both (b) and (c) can obtain improvements due to the representation learning objective achieved by $\mathcal{L}_r^v$ and the clustering objective achieved by $\mathcal{L}_c^v$, respectively. (d) obtains the best performance which indicates that the representation learning objective and the clustering objective have the effect of mutual promotion in our MVCAN, verified their importance.

Two-level multi-view iterative optimization. Figure 1 shows the performance by changing T_1 and T_2 in the first iteration of $\mathcal{T}$-level iteration and $\mathcal{R}$-level iteration. Based on the results, we have the following observations. When $T_1 = 1$ (*i.e.*, the framework is without $\mathcal{T}$-level iteration), MVCAN is unable to infer the scaling factors for different views to generate the more effective robust learning target **T**. Similarly, when $T_2 = 1$ (*i.e.*, the framework is without

$\mathcal{R}$-level iteration), MVCAN cannot learn the more effective representations with the learning target. When T_1 and T_2 increase, the performance also improves, which shows the effectiveness of our two-level multi-view iterative optimization. For all tested datasets, we set $T_1 = 2$ and $T_2 = 100$.

4.4. Model Analysis

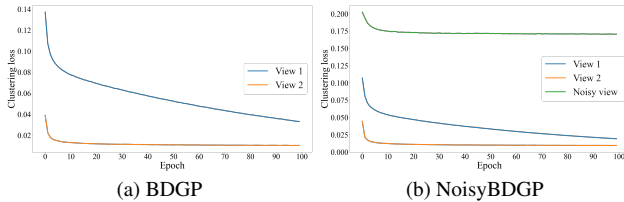This part showcases loss convergence and hyper-parameter analysis to further understand our proposed method.

(a) BDGP

(b) NoisyBDGP

Figure 2. Loss $vs.$ Epoch on BDGP and NoisyBDGP.

(a) ACC $vs.$ λ

(b) NMI $vs.$ λ
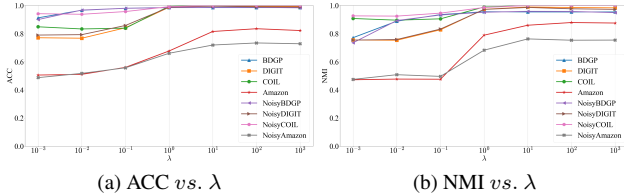
Figure 3. ACC and NMI $vs.$ λ on different datasets.

Loss convergence analysis. Figure 2 plots the clustering loss curve during training and we could observe that the model has good convergence properties. Moreover, it is worth noting that the loss values of the noisy views are larger than that of other views, which is consistent with our analysis in Sec. 2.2. Specifically, the features of the noisy views are not informative and have unclear cluster structures, which makes the clustering loss of noisy views difficult to be minimized. Therefore, we propose to constrain un-shared parameters and inconsistent clustering predictions for multiple views in the multi-view clustering objective of MVCAN, to alleviate the adverse impact of noisy views on the optimization process of other informative views.

Hyper-parameter analysis. The hyper-parameter of MV-CAN includes the trade-off λ in Eq. (11), and Figure 3 shows the clustering effectiveness by traversing λ. The results indicate that λ is insensitive in the range of $[10^1, 10^3]$. Additionally, the cluster number K in the model is changeable. As shown in Figure 4, on DIGIT and NoisyDIGIT, we utilize t-SNE [26] to visualize the scaled representations learned with different cluster numbers. For these two datasets, we mark the representations with ground-truth labels and the truth K is 10. We could observe that MVCAN can learn clear cluster structures on the datasets with noise interference as that on normal ones, indicating the robustness of our method

(a) $K = 5$ (b) $K = 10$ (c) $K = 15$

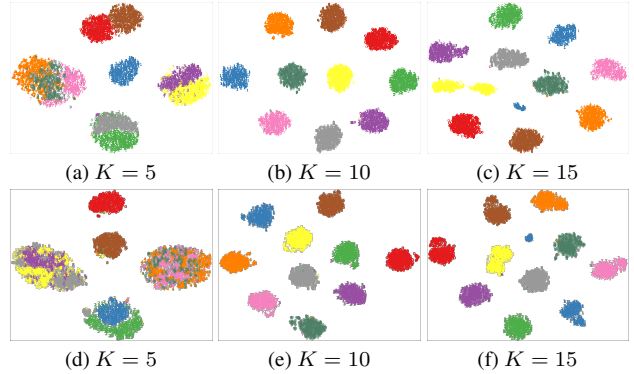(d) $K = 5$ (e) $K = 10$ (f) $K = 15$

Figure 4. Visualization of the representations learned with different prior cluster numbers on DIGIT (a-c) and NoisyDIGIT (d-f).

for noisy views. When K is small ($e.g.$, $K = 5$), we can observe that the representations of digits with similar shapes are gathered together, $e.g.$, "4-7-9" in Figure 4(a). When K is large ($e.g.$, $K = 15$), we observe that the representations of the same digits are separated into two clusters, $e.g.$, "5" in Figure 4(c) (colored in yellow). Consequently, MV-CAN could learn the coarse-grained or fine-grained cluster structures by changing the prior knowledge of K.

5. Conclusion

This paper investigates the pervasive but challenging problem in multi-view clustering, $i.e.$, Noisy-View Drawback (NVD). To mitigate this issue, we proposed a novel deep multi-view clustering method dubbed MVCAN. Comprehensive theoretical and empirical results verified the superior performance of MVCAN, together with the effectiveness of our proposed two conditions in clustering objective and of our two-level multi-view iteration in optimization.

We expect our work to produce beneficial impacts for self-supervised multi-view learning where the information qualities obtained from different views are difficult to be guaranteed and thus they bring noisy information. For example, if the views from some sensors/modalities are faulty or unreliable in unsupervised ensemble environments, it might be promising to take the NVD into account to design algorithms as did in MVCAN. In addition, future work still needs to be devoted to reducing the sensitivity of parameter initialization in deep model and the class number.

References

[1] Mahdi Abavisani and Vishal M Patel. Deep multimodal subspace clustering networks. *IEEE Journal of Selected Topics in Signal Processing*, 12(6):1601–1614, 2018. 14

[2] Pierre Baldi. Autoencoders, unsupervised learning, and deep architectures. In *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, pages 37–49, 2012. 13

[3] Xiao Cai, Hua Wang, Heng Huang, and Chris Ding. Joint stage recognition and anatomical annotation of drosophila gene expression patterns. *Bioinformatics*, 28(12):i16–i24, 2012. 5, 19

[4] Xiaochun Cao, Changqing Zhang, Huazhu Fu, Si Liu, and Hua Zhang. Diversity-induced multi-view subspace clustering. In *CVPR*, pages 586–594, 2015. 1, 12

[5] Jie Chen, Hua Mao, Wai Lok Woo, and Xi Peng. Deep multiview clustering by contrasting cluster assignments. In *ICCV*, pages 16752–16761, 2023. 1

[6] Zhibin Dong, Siwei Wang, Jiaqi Jin, Xinwang Liu, and En Zhu. Cross-view topology based consistent and complementary information for deep multi-view clustering. In *ICCV*, pages 19440–19451, 2023. 1

[7] Shaohua Fan, Xiao Wang, Chuan Shi, Emiao Lu, Ken Lin, and Bai Wang. One2multi graph autoencoder for multi-view graph clustering. In *WWW*, pages 3070–3076, 2020. 14

[8] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *CVPR*, pages 178–178, 2004. 6, 20

[9] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *AISTATS*, pages 315–323, 2011. 20

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, pages 2672–2680, 2014. 14

[11] Xifeng Guo, Long Gao, Xinwang Liu, and Jianping Yin. Improved deep embedded clustering with local structure preservation. In *IJCAI*, pages 1753–1759, 2017. 20

[12] Xiangnan He, Min-Yen Kan, Peichu Xie, and Xiao Chen. Comment-based multi-view clustering of web 2.0 items. In *WWW*, pages 771–782, 2014. 1

[13] Zhenyu Huang, Peng Hu, Joey Tianyi Zhou, Jiancheng Lv, and Xi Peng. Partially view-aligned clustering. In *NeurIPS*, pages 2892–2902, 2020. 13

[14] Zongmo Huang, Yazhou Ren, Xiaorong Pu, Shudong Huang, Zenglin Xu, and Lifang He. Self-supervised graph attention networks for deep weighted multi-view clustering. In *AAAI*, pages 7936–7943, 2023. 2

[15] Jiaqi Jin, Siwei Wang, Zhibin Dong, Xinwang Liu, and En Zhu. Deep incomplete multi-view clustering with cross-view partial sample and prototype alignment. In *CVPR*, pages 11600–11609, 2023. 1, 6, 7, 20

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 20

[17] Ruihuang Li, Changqing Zhang, Huazhu Fu, Xi Peng, Tianyi Zhou, and Qinghua Hu. Reciprocal multi-layer subspace learning for multi-view clustering. In *ICCV*, pages 8172–8180, 2019. 12

[18] Yunfan Li, Dan Zhang, Mouxing Yang, Dezhong Peng, Jun Yu, Yu Liu, Jiancheng Lv, Lu Chen, and Xi Peng. scbridge embraces cell heterogeneity in single-cell rna-seq and atac-seq data integration. *Nature Communications*, 14(1):6045, 2023. 1

[19] Zhaoyang Li, Qianqian Wang, Zhiqiang Tao, Quanxue Gao, and Zhaohua Yang. Deep adversarial multi-view clustering network. In *IJCAI*, pages 2952–2958, 2019. 14

[20] Yijie Lin, Yuanbiao Gou, Zitao Liu, Boyun Li, Jiancheng Lv, and Xi Peng. COMPLETER: Incomplete multi-view clustering via contrastive prediction. In *CVPR*, pages 11174–11183, 2021. 1, 13, 14

[21] Yan-Ching Lin, Min-Chun Hu, Wen-Huang Cheng, Yung-Huan Hsieh, and Hong-Ming Chen. Human action recognition and retrieval using sole depth information. In *ACM MM*, pages 1053–1056, 2012. 6, 20

[22] Jiyuan Liu, Xinwang Liu, Yuexiang Yang, Qing Liao, and Yuanqing Xia. Contrastive multi-view kernel learning. *TPAMI*, pages 1–15, 2023. 1

[23] Xinwang Liu, Xinzhong Zhu, Miaomiao Li, Lei Wang, Chang Tang, Jianping Yin, Dinggang Shen, Huaimin Wang, and Wen Gao. Late fusion incomplete multi-view clustering. *TPAMI*, 41(10):2410–2423, 2018. 1, 12

[24] Xinwang Liu, Li Liu, Qing Liao, Siwei Wang, Yi Zhang, Wenxuan Tu, Chang Tang, Jiyuan Liu, and En Zhu. One pass late fusion multi-view clustering. In *ICML*, pages 6850–6859, 2021. 2, 12, 14

[25] Yiding Lu, Yijie Lin, Mouxing Yang, Dezhong Peng, Peng Hu, and Xi Peng. Decoupled contrastive multi-view clustering with high-order random walks. *arXiv preprint arXiv:2308.11164*, 2023. 1

[26] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using *t*-SNE. *JMLR*, 9:2579–2605, 2008. 8

[27] James MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, 1967. 1, 4

[28] Omid Madani, Manfred Georg, and David Ross. On using nearly-independent feature families for high precision and confidence. *Machine Learning*, 92:457–477, 2013. 6, 20

[29] Sameer A Nene, Shree K Nayar, Hiroshi Murase, et al. Columbia object image library (coil-100). *http://www1. cs. columbia. edu/CAVE/software/softlib/coil-100. php*, 1996. 5, 19

[30] Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *NeurIPS*, pages 849–856, 2001. 1

[31] Feiping Nie, Jing Li, and Xuelong Li. Parameter-free auto-weighted multiple graph learning: a framework for multiview clustering and semi-supervised classification. In *IJCAI*, pages 1881–1887, 2016. 2, 12, 14

[32] Feiping Nie, Lai Tian, and Xuelong Li. Multiview clustering via adaptively weighted procrustes. In *KDD*, pages 2022–2030, 2018. 14, 15

[33] Kamal Nigam and Rayid Ghani. Analyzing the effectiveness and applicability of co-training. In *CIKM*, pages 86–93, 2000. 1

[34] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 1

[35] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. PyTorch: An imperative style, high-performance deep learning library. In *NeurIPS*, pages 8024–8035, 2019. 20

[36] Xi Peng, Zhenyu Huang, Jiancheng Lv, Hongyuan Zhu, and Joey Tianyi Zhou. COMIC: Multi-view clustering without parameter selection. In *ICML*, pages 5092–5101, 2019. 5, 12, 19, 20

[37] Xi Peng, Yunfan Li, Ivor W Tsang, Hongyuan Zhu, Jiancheng Lv, and Joey Tianyi Zhou. Xai beyond classification: Interpretable neural clustering. *JMLR*, 23(1):227–254, 2022. 1

[38] Nimrod Rappoport and Ron Shamir. Multi-omic and multi-view clustering algorithms: review and cancer benchmark. *Nucleic acids research*, 46(20):10546–10562, 2018. 1

[39] Yazhou Ren, Jingyu Pu, Zhimeng Yang, Jie Xu, Guofeng Li, Xiaorong Pu, Philip S Yu, and Lifang He. Deep clustering: A comprehensive survey. *arXiv preprint arXiv:2210.04142*, 2022. 1

[40] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *ECCV*, pages 213–226, 2010. 5, 19

[41] Chang Tang, Zhenglai Li, Jun Wang, Xinwang Liu, Wei Zhang, and En Zhu. Unified one-step multi-view spectral clustering. *TKDE*, 35(6):6449–6460, 2022. 1

[42] Huayi Tang and Yong Liu. Deep safe multi-view clustering: Reducing the risk of clustering performance degradation caused by view increase. In *CVPR*, pages 202–211, 2022. 1, 2, 6, 7, 14

[43] Huayi Tang and Yong Liu. Deep safe incomplete multi-view clustering: Theorem and algorithm. In *ICML*, pages 21090–21110, 2022. 4, 6, 7, 20

[44] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *ECCV*, pages 776–794, 2020. 14

[45] Daniel J. Trosten, Sigurd Løkse, Robert Jenssen, and Michael Kampffmeyer. Reconsidering representation alignment for multi-view clustering. In *CVPR*, pages 1255–1265, 2021. 2, 6, 7, 14

[46] Grigorios Tzortzis and Aristidis Likas. Kernel-based weighted multi-view clustering. In *ICDM*, pages 675–684, 2012. 1, 12

[47] Miguel Ángel Vázquez and Ana I Pérez-Neira. Multigraph spectral clustering for joint content delivery and scheduling in beam-free satellite communications. In *ICASSP*, pages 8802–8806, 2020. 1

[48] Hua Wang, Feiping Nie, and Heng Huang. Multi-view clustering and feature learning via structured sparsity. In *ICML*, pages 352–360, 2013. 12

[49] Qianqian Wang, Zhengming Ding, Zhiqiang Tao, Quanxue Gao, and Yun Fu. Generative partial multi-view clustering with adaptive fusion and cycle consistency. *TIP*, 30:1771–1783, 2021. 1, 2, 4, 6, 7, 14, 20

[50] Siwei Wang, Xinwang Liu, En Zhu, Chang Tang, Jiyuan Liu, Jingtao Hu, Jingyuan Xia, and Jianping Yin. Multi-view clustering via late fusion alignment maximization. In *IJCAI*, pages 3778–3784, 2019. 2, 14

[51] Siwei Wang, Xinwang Liu, Li Liu, Wenxuan Tu, Xinzhong Zhu, Jiyuan Liu, Sihang Zhou, and En Zhu. Highly-efficient incomplete large-scale multi-view clustering with consensus bipartite graph. In *CVPR*, pages 9776–9785, 2022. 1

[52] Yang Wang, Zhang Wenjie, Lin Wu, Xuemin Lin, Meng Fang, and Shirui Pan. Iterative views agreement: an iterative low-rank based structured optimization method to multi-view spectral clustering. In *IJCAI*, pages 2153–2159, 2016. 2, 12, 14

[53] Yang Wang, Lin Wu, Xuemin Lin, and Junbin Gao. Multiview spectral clustering via structured low-rank matrix factorization. *TNNLS*, 29(10):4833–4843, 2018. 2, 12, 14

[54] Jie Wen, Yong Xu, and Hong Liu. Incomplete multiview spectral clustering with adaptive graph learning. *TCYB*, 50 (4):1418–1429, 2018. 12

[55] Jie Wen, Zheng Zhang, Zhao Zhang, Zhihao Wu, Lunke Fei, Yong Xu, and Bob Zhang. DIMC-net: Deep incomplete multi-view clustering network. In *ACM MM*, pages 3753–3761, 2020. 1, 2, 4, 6, 7, 14, 20

[56] Jie Wen, Chengliang Liu, Gehui Xu, Zhihao Wu, Chao Huang, Lunke Fei, and Yong Xu. Highly confident local structure based consensus graph learning for incomplete multi-view clustering. In *CVPR*, pages 15712–15721, 2023. 1

[57] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *ICML*, pages 478–487, 2016. 1, 2, 6, 7, 14, 20

[58] Yuan Xie, Bingqian Lin, Yanyun Qu, Cuihua Li, Wensheng Zhang, Lizhuang Ma, Yonggang Wen, and Dacheng Tao. Joint deep multi-view learning for image clustering. *TKDE*, 33(11): 3594–3606, 2020. 1, 2, 4, 6, 7, 14, 20

[59] Cai Xu, Ziyu Guan, Wei Zhao, Hongchang Wu, Yunfei Niu, and Beilei Ling. Adversarial incomplete multi-view clustering. In *IJCAI*, pages 3933–3939, 2019. 13, 14

[60] Jie Xu, Chao Li, Yazhou Ren, Liang Peng, Yujie Mo, Xiaoshuang Shi, and Xiaofeng Zhu. Deep incomplete multi-view clustering via mining cluster complementarity. In *AAAI*, pages 8761–8769, 2022. 1, 2, 6, 7, 13, 14, 20

[61] Jie Xu, Yazhou Ren, Xiaoshuang Shi, Heng Tao Shen, and Xiaofeng Zhu. UNTIE: Clustering analysis with disentanglement in multi-view information fusion. *Information Fusion*, 100:101937, 2023. 20

[62] Jie Xu, Yazhou Ren, Huayi Tang, Zhimeng Yang, Lili Pan, Yang Yang, Xiaorong Pu, Philip S. Yu, and Lifang He. Self-supervised discriminative feature learning for deep multi-view clustering. *TKDE*, 35(7):7470–7482, 2023. 1, 4, 6, 7, 14

[63] Weiqing Yan, Yuanyang Zhang, Chenlei Lv, Chang Tang, Guanghui Yue, Liang Liao, and Weisi Lin. GCFAgg: Global and cross-view feature aggregation for multi-view clustering. In *CVPR*, pages 19863–19872, 2023. 1

[64] Mouxing Yang, Yunfan Li, Peng Hu, Jinfeng Bai, Jiancheng Lv, and Xi Peng. Robust multi-view clustering with incomplete information. *TPAMI*, 45(1):1055–1069, 2022. 1, 13

[65] Zuyuan Yang, Naiyao Liang, Wei Yan, Zhenni Li, and Shengli Xie. Uniform distribution non-negative matrix factorization for multiview clustering. *TCYB*, pages 3249–3262, 2021. 12

[66] Yongkai Ye, Xinwang Liu, and Jianping Yin. Multi-view clustering with noisy views. In *Proceedings of the International Conference on Computer Science and Artificial Intelligence*, pages 339–344, 2018. 2, 7, 14

[67] Kun Zhan, Feiping Nie, Jing Wang, and Yi Yang. Multiview consensus graph clustering. *TIP*, 28(3):1261–1270, 2018. 1, 12

[68] Kun Zhan, Chaoxi Niu, Changlu Chen, Feiping Nie, Changqing Zhang, and Yi Yang. Graph structure fusion for multiview clustering. *TKDE*, 31(10):1984–1993, 2019. 2, 14

[69] Changqing Zhang, Qinghua Hu, Huazhu Fu, Pengfei Zhu, and Xiaochun Cao. Latent multi-view subspace clustering. In *CVPR*, pages 4279–4287, 2017. 2, 12, 14

[70] Changqing Zhang, Huazhu Fu, Qinghua Hu, Xiaochun Cao, Yuan Xie, Dacheng Tao, and Dong Xu. Generalized latent multi-view subspace clustering. *TPAMI*, 42(1):86–99, 2018. 1

[71] Changqing Zhang, Yajie Cui, Zongbo Han, Joey Tianyi Zhou, Huazhu Fu, and Qinghua Hu. Deep partial multi-view learning. *TPAMI*, 44(5):2402–2415, 2020. 1

[72] Runwu Zhou and Yi-Dong Shen. End-to-end adversarial-attention network for multi-modal clustering. In *CVPR*, pages 14619–14628, 2020. 2, 6, 14, 20

[73] Tao Zhou, Changqing Zhang, Xi Peng, Harish Bhaskar, and Jie Yang. Dual shared-specific multiview subspace clustering. *TCYB*, 50(8):3517–3530, 2019. 2, 12, 14

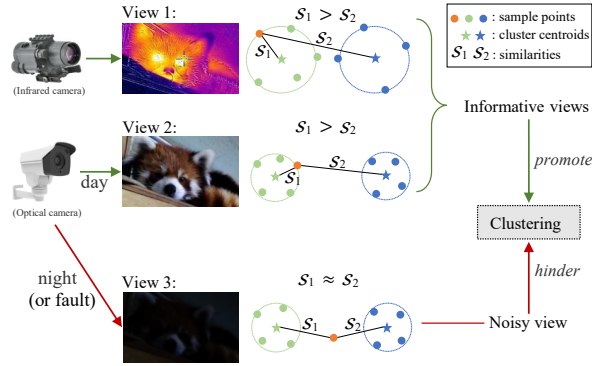# Appendix A: Framework, Related Work, and Notations

Figure 1. Illustration of the noisy-view drawback (NVD). The informative views have distinct representation similarities, which can promote clustering due to their consistency and complementarity. However, the noisy views have indistinct representation similarities. For instance, the views extracted from faulty or inapplicable sensors will bring noisy information and hinder clustering, making it be of practical significance to investigate the noise robustness.
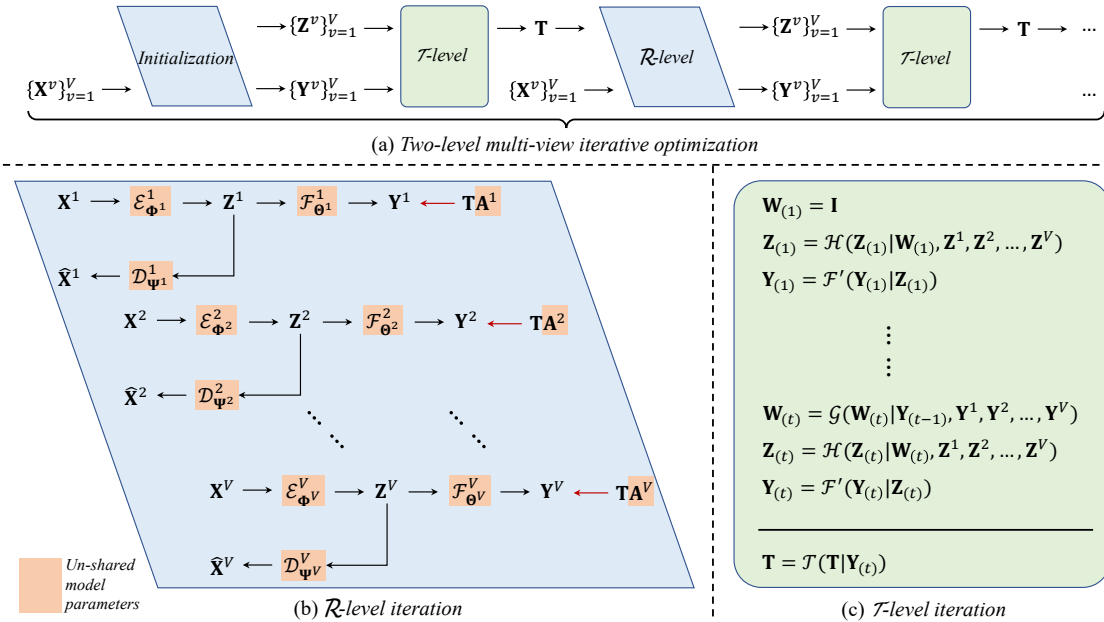

Figure 2. The frame diagram of our MVCAN. Specifically, (a) MVCAN utilizes the two-level multi-view iterative optimization strategy to train the model for clustering multi-view data $\{\mathbf{X}^v\}_{v=1}^V$, where (b) $\mathcal{R}$-level iteration adjusts $\mathbf{T}$ for corresponding to each $\mathbf{Y}^v$ by $\mathbf{A}^v$ and updates the decoupled model with un-shared network parameters for V views, to obtain their individual representations $\{\mathbf{Z}^v\}_{v=1}^V$ and clustering soft labels $\{\mathbf{Y}^v\}_{v=1}^V$; (c) $\mathcal{T}$-level iteration is established to infer the robust learning target $\mathbf{T}$ based on the already learned $\{\mathbf{Z}^v\}_{v=1}^V$ and $\{\mathbf{Y}^v\}_{v=1}^V$ (including the iterations of scaling matrix $\mathbf{W}_{(t)}$, scaled representation $\mathbf{Z}_{(t)}$, and robust soft labels $\mathbf{Y}_{(t)}$). *Note:* $\mathbf{W}_{(t)}$ plays the role of scaling the values of different dimensions of $\mathbf{Z}_{(t)}$, so we name it the scaling matrix. $\mathbf{Y}_{(t)}$ finally predicts the clustering results.

In the literature, existing multi-view clustering (MVC) methods could be divided into two groups, *i.e.*, traditional methods and deep methods. In this paper, we briefly introduce traditional MVC methods and focus on deep MVC methods as follows.

Traditional MVC methods learn the representations of multi-view data for clustering by leveraging the classical machine learning technologies, such as graph MVC [36, 54, 67], subspace MVC [4, 17, 73], kernel MVC [23, 24, 46], and matrix factorization MVC [53, 65]. Typically, many traditional MVC methods are limited by their representation capability of shallow models such that they usually perform clustering tasks with the limited feature forms and data scales [31, 48, 52, 69].

Table 1. Notations and their descriptions in this paper.

Notations	Descriptions
N	the number of samples or the data size
V	the number of views in the multi-view dataset
K	the number of clusters in the multi-view dataset
i,j,k,v,t,e,m^*	the index notations
$\mathbf{X}$	the data for a single-view method, $\mathbf{x}_i \in \mathbf{X}$
$\mathbf{Z}$	the learned representation in a single-view method, $\mathbf{z}_i \in \mathbf{Z}$
$\mathbf{Y}$	the learned soft labels in a single-view method, $y_{ij} \in \mathbf{Y}$
$\mathbf{T}$	the learning target, $\mathbf{t}_i \in \mathbf{T}$, $t_{ij} \in \mathbf{T}$
$\mathbf{X}^v$	the v-th view's data for a multi-view method, $\mathbf{x}_i^v \in \mathbf{X}^v$
$\mathbf{Z}^v$	the v-th view's representations learned by a multi-view method, $\mathbf{z}_i^v \in \mathbf{Z}^v$
$\mathbf{Y}^v$	the v-th view's soft labels in a multi-view method, $\mathbf{y}_i^v \in \mathbf{Y}^v$, $y_{ij}^v \in \mathbf{Y}^v$
$\hat{\mathbf{X}}^v$	the v-th view's reconstructed data for a multi-view method
D_v	the dimensionality of $\mathbf{X}^v$ and $\hat{\mathbf{X}}^v$
d_v	the dimensionality of $\mathbf{Z}^v$
$\mathcal{E}_{\mathbf{\Phi}}$	the encoder with parameter $\mathbf{\Phi}$ for a single-view method
$\boldsymbol{\mu}_j$	the j-th cluster centroid in a single-view method
$\mathcal{E}_{\mathbf{\Phi}^v}^v$	the v-th view's encoder with parameter $\mathbf{\Phi}^v$ for a multi-view method
$\mathcal{D}_{\mathbf{\Psi}^v}^v$	the v-th view's decoder with parameter $\mathbf{\Psi}^v$ for a multi-view method
$\boldsymbol{\mu}_j^v$	the j-th cluster centroid in the v-th view for a multi-view method
$\mathcal{F}_{\mathbf{\Theta}}$	the fusion module $\mathcal{F}$ with the parameter $\mathbf{\Theta}$ shared for multiple views
$\mathcal{F}_{\mathbf{\Theta}^v}^v$	the v-th view's clustering module $\mathcal{F}^v$ with the parameter $\mathbf{\Theta}^v$ in MVCAN
$\mathcal{D}(\mathbf{a}, \mathbf{b})$	the squared Euclidean distance between representations $\mathbf{a}$ and $\mathbf{b}$
$\mathcal{S}(\mathbf{a}, \mathbf{b})$	the similarity between representations $\mathbf{a}$ and $\mathbf{b}$
ε	a sufficiently small value in Definitions
δ	a threshold in Theorems
$\mathbf{L}$	the ground-truth label matrix
$\mathbf{A}$	the matching matrix to calculate the clustering accuracy
$\check{\mathbf{Y}}$	the transformed prediction label matrix, $\check{\mathbf{y}}_i \in \check{\mathbf{Y}}$, $\check{y}_{ij} \in \check{\mathbf{Y}}$
$\mathbf{A}^v$	the matching matrix in the Condition 2 of MVCAN
$\mathbf{I}, \mathbf{I}_K, \mathbf{I}^v$	the unit matrices
$\mathbf{W}_{(t)}$	the scaling matrix in MVCAN, $w_{(t)}^v \in \mathbf{W}_{(t)}$
$\mathbf{Z}_{(t)}$	the scaled representation in MVCAN, $\mathbf{z}_{i(t)} \in \mathbf{Z}_{(t)}$
$\mathbf{Y}_{(t)}$	the robust soft labels in MVCAN, $y_{ij(t)} \in \mathbf{Y}_{(t)}$
$\mathbf{c}_{j(t)}$	the j-th cluster centroid of $\mathbf{Z}_{(t)}$ in the t-th iteration
$\mathcal{L}_r^v$	the representation learning objective of the v-th view in MVCAN
$\mathcal{L}_c^v$	the clustering objective of the v-th view in MVCAN
$\mathcal{L}_K$	the clustering objective of K-means
$\mathcal{L}^v$	the loss function to train the deep model of the v-th view in the $\mathcal{R}$-level iteration of MVCAN
λ	the hyper-parameter to achieve the trade-off between $\mathcal{L}_r^v$ and $\mathcal{L}_c^v$
T_1	the iteration number in the $\mathcal{T}$-level iterative optimization
T_2	the iteration number in the $\mathcal{R}$-level iterative optimization
E	the number of training epochs
$\arg\max_j y_{ij(t)}$	the final cluster assignment for the i-th sample based on the robust soft label $y_{ij(t)} \in \mathbf{Y}_{(t)}$

During past years, many deep MVC methods have been proposed [13, 59, 64]. The mainstream techniques used in deep MVC methods belong to self-supervised learning. Deep autoencoder [2] is the most poplar model used for deep MVC [20, 59, 60], which conducts data reconstruction processes and can create basic self-supervision signals for unsupervised clustering tasks. Some deep MVC methods are based on the aforementioned machine learning technologies, such as deep

subspace MVC [1] and deep graph MVC [7]. These methods usually leverage the representation learning capability of deep autoencoder networks as well as the data mining capability of traditional technologies. Additionally, many work investigate various deep learning technologies to develop deep MVC methods. For example, some work [19, 49, 59, 72] combine GAN [10] with clustering objective for multiple views. Some work [20, 45] introduce the contrastive learning [20, 44] to learn the consistency of multi-view representations for clustering. Establishing clustering pseudo labels is an important approach to achieve end-to-end MVC. In research work, one of the most popular learning paradigms in deep MVC is based on the single-view clustering (SVC) method with self-training, entitled deep embedded clustering (DEC [57]), *e.g.*, [7, 49, 55, 58, 59, 62], which establishes pseudo labels and then produces self-supervised signals to learn the clustering-oriented representations. Compared with traditional methods, deep MVC methods usually have superior representation capability and scalability.

In practical multi-view scenarios, sample features extracted from some views could be noisy and might have harmful information for MVC. Although some efforts take the view quality into account and propose weighting strategies in the fusion of multiple views [32, 45, 49, 55, 66], the low-quality views could be treated as a special case of noise and both the noisy views and the low-quality views will make it difficult to uncover the effective cluster patterns for most existing MVC methods. Because the model network parameters in fusion methods [24, 45, 49, 50, 52, 55, 69, 73] usually are shared for multiple views, and many methods hope to obtain consistent clustering predictions for different views [31, 32, 42, 53, 60, 62, 66, 68, 72], the noisy-view drawback (NVD) is easy to make the learning of all views degenerate, and thus losing the useful consistency and complementarity among multiple views. The work most similar to ours are the precursor deep MVC studies [60, 62], which construct a consistent self-supervised clustering objective for multiple views and design a weighted concatenation of multi-view features, respectively. As NVD might cause that MVC is not necessarily better than SVC and limit the application of MVC methods, it is a meaningful but challenging goal to address the NVD and we require ongoing solutions. In this paper, we modify the multi-view clustering objective to allow for inconsistent clustering results across different views in parameter-decoupled models, and propose an iterative approach to learn the robust self-supervised target for addressing NVD.

Appendix B: Theoretical Analysis and Proofs

Definition 1. *Denoting $\mathcal{D}(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\|_2^2$ as squared Euclidean distance between the representations $\mathbf{a}$ and $\mathbf{b}$,*

$$\mathcal{S}(\mathbf{a}, \mathbf{b}) := \frac{1}{1 + \mathcal{D}(\mathbf{a}, \mathbf{b})} \in (0, 1]$$

is defined as the representation similarity between $\mathbf{a}$ and $\mathbf{b}$. Formally, $y_{ij}^v \in (0, 1]$ holds given Eq. (1).

Letting $\boldsymbol{\mu}_a^v$ and $\boldsymbol{\mu}_b^v$ denote the a-th and the b-th cluster centroids in the representation space of $\mathbf{Z}^v$, $\mathbf{z}_i^v \in \mathbf{Z}^v$, we have $\mathcal{S}(\mathbf{z}_i^v, \boldsymbol{\mu}_a^v) > \mathcal{S}(\mathbf{z}_i^v, \boldsymbol{\mu}_b^v) \Rightarrow y_{ia}^v > y_{ib}^v$ and $\mathcal{S}(\mathbf{z}_i^v, \boldsymbol{\mu}_a^v) < \mathcal{S}(\mathbf{z}_i^v, \boldsymbol{\mu}_b^v) \Rightarrow y_{ia}^v < y_{ib}^v$, because

$$
\begin{aligned}
y_{ij}^v &= \frac{(1 + \|\mathbf{z}_i^v - \boldsymbol{\mu}_j^v\|_2^2)^{-1}}{\sum_{j=1}^K (1 + \|\mathbf{z}_i^v - \boldsymbol{\mu}_j^v\|_2^2)^{-1}} \\
&= \frac{\mathcal{S}(\mathbf{z}_i^v, \boldsymbol{\mu}_j^v)}{\sum_{j=1}^K \mathcal{S}(\mathbf{z}_i^v, \boldsymbol{\mu}_j^v)} \\
&\propto \mathcal{S}(\mathbf{z}_i^v, \boldsymbol{\mu}_j^v).
\end{aligned}
$$

Definition 2. *(ε, $\mathbf{z}$, $\boldsymbol{\mu}$ - Noisy-view) For $\forall \mathbf{z}_i^v \in \mathbf{Z}^v$, it belongs to the noisy view if $\exists \boldsymbol{\mu}_a^v$, $\boldsymbol{\mu}_b^v$, and $\varepsilon > 0$ such that $|\mathcal{D}(\mathbf{z}_i^v, \boldsymbol{\mu}_a^v) - \mathcal{D}(\mathbf{z}_i^v, \boldsymbol{\mu}_b^v)| < \varepsilon$, $\mathcal{S}(\mathbf{z}_i^v, \boldsymbol{\mu}_a^v) \approx \mathcal{S}(\mathbf{z}_i^v, \boldsymbol{\mu}_b^v)$, and $y_{ia}^v \approx y_{ib}^v$, where ε is a sufficiently small value. Otherwise, $\mathbf{z}_i^v$ is the informative view.*

Theorem 1. *Denoting $\check{\mathbf{Y}} = \mathbf{LA}$, where $\mathbf{A} \in \{0, 1\}^{K \times K}$ makes $\check{\mathbf{Y}}$ maximally match the learning target $\mathbf{T}$. Then, the clustering accuracy can be calculated as $ACC = \frac{1}{N}\left(N - \frac{1}{2}\|\check{\mathbf{Y}} - \mathbf{T}\|_F^2\right) = 1 - \frac{1}{2N}\|\check{\mathbf{Y}} - \mathbf{T}\|_F^2$. In Eq. (3), if $\boldsymbol{\Theta}$ is shared by multiple views and their soft labels $\{\mathbf{Y}^v\}_{v=1}^V$ have consistent learning target $\mathbf{T}$, we have*

$$ACC \leq 1 - \frac{1}{2N}\left(\left(\max_{1 \leq m \leq V}\|\check{\mathbf{Y}} - \mathcal{F}_{\boldsymbol{\Theta}}(\mathbf{Y}^m|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2\right) - \|\mathbf{T} - \mathcal{F}_{\boldsymbol{\Theta}}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2\right).$$

Proof. Motivated by the proof in [32], we first consider the following equation:

$$\|\check{\mathbf{Y}} - \mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2 - \|\mathbf{T} - \mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2$$

$$=\|\check{\mathbf{Y}}\|_F^2 - \|\mathbf{T}\|_F^2 + \|\mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2 - \|\mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2 - 2Tr\left((\check{\mathbf{Y}} - \mathbf{T})^T \mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\right)$$

$$= (N - \|\mathbf{T}\|_F^2) + \left(\|\mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2 - \|\mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2\right) - 2Tr\left((\check{\mathbf{Y}} - \mathbf{T})^T \mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\right)$$

$$= (N - \|\mathbf{T}\|_F^2) + 2Tr\left((\mathbf{T} - \check{\mathbf{Y}})^T \mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\right)$$

$$= (N - \|\mathbf{T}\|_F^2) + 2Tr\left(\mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)(\mathbf{T} - \check{\mathbf{Y}})^T\right),$$

where the term of matrix trace could satisfy the inequality:

$$2Tr\left(\mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)(\mathbf{T} - \check{\mathbf{Y}})^T\right)$$

$$=2\sum_{i=1}^N \mathbf{y}_i^v(\mathbf{t}_i^T - \check{\mathbf{y}}_i^T)$$

$$=2\sum_{i=1}^N \mathbf{y}_i^v\mathbf{t}_i^T - \mathbf{y}_i^v\check{\mathbf{y}}_i^T$$

$$=2\sum_{\mathbf{t}_i \neq \check{\mathbf{y}}_i} \left(\sum_{k=1}^K y_{ik}^v t_{ik}\right) - \left(\sum_{k=1}^K y_{ik}^v \check{y}_{ik}\right)$$

$$\leq 2\sum_{\mathbf{t}_i \neq \check{\mathbf{y}}_i} \left|\left(\sum_{k=1}^K y_{ik}^v t_{ik}\right) - \left(\sum_{k=1}^K y_{ik}^v \check{y}_{ik}\right)\right|$$

$$\leq 2\sum_{\mathbf{t}_i \neq \check{\mathbf{y}}_i} |1 - 0|$$

$$=2\sum_{\mathbf{t}_i \neq \check{\mathbf{y}}_i} 1$$

$$=\|\check{\mathbf{Y}} - \mathbf{T}\|_F^2.$$

Then, we have

$$\|\check{\mathbf{Y}} - \mathbf{T}\|_F^2 \geq \|\check{\mathbf{Y}} - \mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2 - \|\mathbf{T} - \mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2 - N + \|\mathbf{T}\|_F^2$$

$$\geq \left(\max_{1 \leq m \leq V} \|\check{\mathbf{Y}} - \mathcal{F}_{\Theta}(\mathbf{Y}^m|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2\right) - \|\mathbf{T} - \mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2 - N + \left(\max_{\mathbf{T}} \|\mathbf{T}\|_F^2\right)$$

$$\geq \left(\max_{1 \leq m \leq V} \|\check{\mathbf{Y}} - \mathcal{F}_{\Theta}(\mathbf{Y}^m|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2\right) - \|\mathbf{T} - \mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2.$$

Furthermore, the clustering accuracy becomes

$$ACC = \frac{1}{N}\left(N - \frac{1}{2}\|\check{\mathbf{Y}} - \mathbf{T}\|_F^2\right) = 1 - \frac{1}{2N}\|\check{\mathbf{Y}} - \mathbf{T}\|_F^2$$

$$\leq 1 - \frac{1}{2N}\left(\left(\max_{1 \leq m \leq V} \|\check{\mathbf{Y}} - \mathcal{F}_{\Theta}(\mathbf{Y}^m|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2\right) - \|\mathbf{T} - \mathcal{F}_{\Theta}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2\right),$$

which completes this proof. □

A specific example for illustrating Theorem 1. Given the DEC framework as a specific example, we denote $\mathbf{Z}^m$ and $\mathbf{Z}^n$, respectively, as the representations of the informative view and the noisy/low-quality view. The shared parameters are

the K cluster centroids $\mathbf{U} = [\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_K]$, which are the common cluster centroids for the all views' representations. Based on $\mathbf{Z}^m$ and $\mathbf{U}$ as formulated in Eq. (1), the informative view's representation $\mathbf{Z}^m$ has clear cluster structures and thus its clustering loss $\mathcal{L}_c^m$ is easy to minimize. However, based on $\mathbf{Z}^n$ and the same $\mathbf{U}$ as formulated in Eq. (1), the noisy/low-quality view's representation $\mathbf{Z}^n$ has unclear cluster structures and thus its clustering loss $\mathcal{L}_c^n$ is hard to minimize. On the one hand, since the training objectives of multiple views have the same learning target $\mathbf{T}$, the large $\mathcal{L}_c^n$ will dominate the small $\mathcal{L}_c^m$ if their optimizations are not decoupled. On the other hand, the minimization of the noisy view's clustering loss $\mathcal{L}_c^n$ will change the shared K cluster centroids $\mathbf{U}$, causing that $\mathbf{U}$ is not suitable for the informative view's representations $\mathbf{Z}^m$. As a result, the NVD of low-quality or noisy views destroys the method's robustness.

Theorem 2. *Denoting $\mathcal{L}_K$ as the K-means objective, $\mathcal{L}_K(\mathbf{Z}_{(t)})$ is equivalent to punishing different scaling factors on $\{\mathcal{L}_K(\mathbf{Z}^v)\}_{v=1}^V$ under the consistency constraint of multiple views' cluster centroids.*

Proof. For clarity, we may omit the iteration symbol (t) of $\mathbf{Z}_{(t)}$, $\mathbf{W}_{(t)}$, and $\mathbf{c}_{j(t)}$ in next proofs. Then, the K-means objective on $\mathbf{Z}$ (*i.e.*, $\mathbf{Z}_{(t)}$) can be formulated as:

$$
\begin{aligned}
\mathcal{L}_K(\mathbf{Z}) &= \min_{\mathbf{c}_j} \sum_{i=1}^N \sum_{j=1}^K \|\mathbf{z}_i - \mathbf{c}_j\|_2^2 \\
&= \min_{\boldsymbol{\mu}_j^v} \sum_{i=1}^N \sum_{j=1}^K \sum_{v=1}^V \|w^v \mathbf{z}_i^v - w^v \boldsymbol{\mu}_j^v\|_2^2, \, s.t. \, \mathbf{c}_j = \begin{bmatrix} \boldsymbol{\mu}_j^1 & \boldsymbol{\mu}_j^2 & \cdots & \boldsymbol{\mu}_j^V \end{bmatrix} \mathbf{W} \\
&= \sum_{v=1}^V (w^v)^2 \min_{\boldsymbol{\mu}_j^v} \sum_{i=1}^N \sum_{j=1}^K \|\mathbf{z}_i^v - \boldsymbol{\mu}_j^v\|_2^2, \, s.t. \, \mathbf{c}_j = \begin{bmatrix} \boldsymbol{\mu}_j^1 & \boldsymbol{\mu}_j^2 & \cdots & \boldsymbol{\mu}_j^V \end{bmatrix} \mathbf{W} \\
&= \sum_{v=1}^V (w^v)^2 \mathcal{L}_K(\mathbf{Z}^v), \, s.t. \, \mathbf{c}_j = \begin{bmatrix} \boldsymbol{\mu}_j^1 & \boldsymbol{\mu}_j^2 & \cdots & \boldsymbol{\mu}_j^V \end{bmatrix} \mathbf{W},
\end{aligned}
$$

where $\mathbf{c}_j$ and $\boldsymbol{\mu}_j^v$ denote the j-th cluster centroid of $\mathbf{Z}$ and $\mathbf{Z}^v$, respectively. First, we can find that $(w^v)^2 \mathcal{L}_K(\mathbf{Z}^v)$ punishes different scaling factors on different views. Second, if the K-means objective is conducted on $\mathbf{Z}^v$ individually, the obtained cluster centroids for the same cluster of two views might have different samples, *e.g.*, $\boldsymbol{\mu}_j^a = \sum_{i \in \Omega^a} \mathbf{z}_i^a / |\Omega^a|$ and $\boldsymbol{\mu}_j^b = \sum_{i \in \Omega^b} \mathbf{z}_i^b / |\Omega^b|$ where $\Omega^a \neq \Omega^b$. However, the constraint condition $\mathbf{c}_j = \begin{bmatrix} \boldsymbol{\mu}_j^1 & \boldsymbol{\mu}_j^2 & \cdots & \boldsymbol{\mu}_j^V \end{bmatrix} \mathbf{W}$ makes $\Omega^a = \Omega^b = \Omega^c$ such that $w^a \boldsymbol{\mu}_j^a = \sum_{i \in \Omega^a} w^a \mathbf{z}_i^a / |\Omega^a|$, $w^b \boldsymbol{\mu}_j^b = \sum_{i \in \Omega^b} w^b \mathbf{z}_i^b / |\Omega^b|$, and $\mathbf{c}_j = \sum_{i \in \Omega^c} \mathbf{z}_i / |\Omega^c|$ for $\forall a, b \in \{1, 2, \ldots, V\}$. Therefore, the constraint condition could be treated as the consistency constraint of multiple views' cluster centroids, which guarantees that the cluster centroids $\mathbf{c}_j, \boldsymbol{\mu}_j^1, \boldsymbol{\mu}_j^2, \ldots, \boldsymbol{\mu}_j^V$ are obtained from the consistent samples among multiple views. $\qquad\square$

Theorem 3. *(Consistency) If a sample representation is informative in multiple views and has the same cluster assignments in these views, its cluster assignment in $\mathbf{Y}_{(t)}$ is the same as that in these views.*

Proof. For clarity, we take two views as example, $\mathbf{z}_i = \begin{bmatrix} w^1 \mathbf{z}_i^1 & w^2 \mathbf{z}_i^2 \end{bmatrix} \in \mathbf{Z}$ and we let $\mathbf{c}_j = \begin{bmatrix} w^1 \boldsymbol{\mu}_j^1 & w^2 \boldsymbol{\mu}_j^2 \end{bmatrix}$ denote the ideal cluster centroid in the scaled representation space of $\mathbf{Z}$, where $\boldsymbol{\mu}_j^1$ and $\boldsymbol{\mu}_j^2$ are the cluster centroids in the 1-st view and the 2-nd view of the j-th cluster, respectively. For any two clusters, *e.g.*, $\mathbf{c}_1$ and $\mathbf{c}_2$, we have

$$
\begin{aligned}
\mathcal{D}(\mathbf{z}_i, \mathbf{c}_1) &= \left\| \begin{bmatrix} w^1 \mathbf{z}_i^1 & w^2 \mathbf{z}_i^2 \end{bmatrix} - \begin{bmatrix} w^1 \boldsymbol{\mu}_1^1 & w^2 \boldsymbol{\mu}_1^2 \end{bmatrix} \right\|_2^2 \\
&= \|w^1 \mathbf{z}_i^1 - w^1 \boldsymbol{\mu}_1^1\|_2^2 + \|w^2 \mathbf{z}_i^2 - w^2 \boldsymbol{\mu}_1^2\|_2^2 \\
&= (w^1)^2 \|\mathbf{z}_i^1 - \boldsymbol{\mu}_1^1\|_2^2 + (w^2)^2 \|\mathbf{z}_i^2 - \boldsymbol{\mu}_1^2\|_2^2 \\
&= (w^1)^2 \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) + (w^2)^2 \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2).
\end{aligned}
$$

Similarly, $\mathcal{D}(\mathbf{z}_i, \mathbf{c}_2) = (w^1)^2 \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) + (w^2)^2 \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2)$. Furthermore, we have

$$
\begin{aligned}
\mathcal{D}(\mathbf{z}_i, \mathbf{c}_1) - \mathcal{D}(\mathbf{z}_i, \mathbf{c}_2) &= (w^1)^2 \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) + (w^2)^2 \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - (w^1)^2 \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) - (w^2)^2 \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) \\
&= (w^1)^2 \left(\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) \right) + (w^2)^2 \left(\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) \right).
\end{aligned}
$$

16

If $y_{i1}^1 > y_{i2}^1$ and $y_{i1}^2 > y_{i2}^2$, the sample is informative in both views and has the same cluster assignment, *i.e.*, $\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) < \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1)$ and $\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) < \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2)$. Then, the following inequality holds:

$$\mathcal{D}(\mathbf{z}_i, \mathbf{c}_1) - \mathcal{D}(\mathbf{z}_i, \mathbf{c}_2) = (w^1)^2 \left(\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) \right) + (w^2)^2 \left(\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) \right) < 0,$$

which indicates that $y_{i1} > y_{i2}$. Similarly, if $y_{i1}^1 < y_{i2}^1$ and $y_{i1}^2 < y_{i2}^2$, we have $\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) > \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1)$, $\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) > \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2)$, and $\mathcal{D}(\mathbf{z}_i, \mathbf{c}_1) - \mathcal{D}(\mathbf{z}_i, \mathbf{c}_2) > 0$ which indicates that $y_{i1} < y_{i2}$. As a result, the sample has the same cluster assignment in $\mathbf{Y}_{(t)}$ as that in the informative views, *i.e.*, $\mathbf{Y}^1$ and $\mathbf{Y}^2$. $\qquad\square$

Theorem 4. *(**Complementarity**) If a sample representation is informative in multiple views where it has different cluster assignments, we have two cases according to the differences of similarity among the clusters.*
Case 1: if the differences of similarity among the clusters are equal, its cluster assignment in $\mathbf{Y}_{(t)}$ is the same as that in the informative view with the largest scaling factor.
Case 2: if the differences of similarity among the clusters are not equal, its cluster assignment in $\mathbf{Y}_{(t)}$ is more likely to be the same as that in the informative view with the largest scaling factor.

Proof. Letting $\Delta^v = |\mathcal{D}(\mathbf{z}_i^v, \boldsymbol{\mu}_1^v) - \mathcal{D}(\mathbf{z}_i^v, \boldsymbol{\mu}_2^v)|$, and $\Gamma^v = |\mathcal{S}(\mathbf{z}_i^v, \boldsymbol{\mu}_1^v) - \mathcal{S}(\mathbf{z}_i^v, \boldsymbol{\mu}_2^v)|$ denote the difference of similarity, we have $\Delta^1 = \Delta^2 \Rightarrow |\mathcal{S}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{S}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1)| = |\mathcal{S}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - \mathcal{S}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2)|$.

Case 1: if $(\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1))(\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2)) < 0$ and $\Delta^1 = \Delta^2$, the sample is informative in both views but it has different cluster assignments (*i.e.*, $\arg\max_j y_{ij}^1 \neq \arg\max_j y_{ij}^2$), and the differences of similarity are equal $\Gamma^1 = \Gamma^2$. In this case, we obtain a boundary condition:

$$\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) = \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2),$$

and thus $\mathcal{D}(\mathbf{z}_i, \mathbf{c}_1) - \mathcal{D}(\mathbf{z}_i, \mathbf{c}_2)$ in Theorem 3 becomes

$$\begin{aligned}
\mathcal{D}(\mathbf{z}_i, \mathbf{c}_1) - \mathcal{D}(\mathbf{z}_i, \mathbf{c}_2) &= (w^1)^2 \left(\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) \right) + (w^2)^2 \left(\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) \right) \\
&= ((w^1)^2 - (w^2)^2)(\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1)).
\end{aligned}$$

If $w^1 > w^2$, we further have

$$\begin{aligned}
y_{i1}^1 > y_{i2}^1 &\Rightarrow \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) < \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) \\
&\Rightarrow \mathcal{D}(\mathbf{z}_i, \mathbf{c}_1) < \mathcal{D}(\mathbf{z}_i, \mathbf{c}_2) \\
&\Rightarrow y_{i1} > y_{i2}
\end{aligned}$$

and $y_{i1}^1 < y_{i2}^1 \Rightarrow y_{i1} < y_{i2}$.
If $w^1 < w^2$, we similarly have

$$\begin{aligned}
y_{i1}^2 > y_{i2}^2 &\Rightarrow \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) < \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) \\
&\Rightarrow' \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) > \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) \\
&\Rightarrow \mathcal{D}(\mathbf{z}_i, \mathbf{c}_1) < \mathcal{D}(\mathbf{z}_i, \mathbf{c}_2) \\
&\Rightarrow y_{i1} > y_{i2}
\end{aligned}$$

and $y_{i1}^2 < y_{i2}^2 \Rightarrow y_{i1} < y_{i2}$. $\Rightarrow'$ holds because $(\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1))(\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2)) < 0$. In conclusion, the sample has the same cluster assignment in $\mathbf{Y}_{(t)}$ as that the view with the largest scaling factor.

Case 2: if $(\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1))(\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2)) < 0$ and $\Delta^1 \neq \Delta^2$, the sample is informative in both views but it has different cluster assignments (*i.e.*, $\arg\max_j y_{ij}^1 \neq \arg\max_j y_{ij}^2$), and the differences of similarity are not equal $\Gamma^1 \neq \Gamma^2$.

In this case, if $y_{i1}^1 < y_{i2}^1$ and $y_{i1}^2 > y_{i2}^2$, *i.e.*, $\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) > \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1)$ and $\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) < \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2)$, there is a threshold δ as $\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) \neq \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1)$ and $\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) \neq \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2)$:

$$\delta = \frac{\Delta^2}{\Delta^1} = \frac{\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2)}{\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1)}.$$

When w^1 is large so that $\frac{(w^1)^2}{(w^2)^2} > \delta$, we have

$$\frac{(w^1)^2}{(w^2)^2} > \frac{\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2)}{\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1)}$$
$$\Rightarrow (w^1)^2 \left(\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) \right) + (w^2)^2 \left(\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) \right) > 0$$
$$\Rightarrow \mathcal{D}(\mathbf{z}_i, \mathbf{c}_1) - \mathcal{D}(\mathbf{z}_i, \mathbf{c}_2) > 0$$
$$\Rightarrow y_{i1} < y_{i2},$$

which indicates $y_{i1}^1 < y_{i2}^1 \Rightarrow y_{i1} < y_{i2}$, *i.e.*, the sample has the same cluster assignment in $\mathbf{Y}_{(t)}$ as that in the view with the scaling factor w^1.

When w^2 is large so that $\frac{(w^1)^2}{(w^2)^2} < \delta$, we have

$$\frac{(w^1)^2}{(w^2)^2} < \frac{\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2)}{\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1)}$$
$$\Rightarrow (w^1)^2 \left(\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) \right) + (w^2)^2 \left(\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) \right) < 0$$
$$\Rightarrow \mathcal{D}(\mathbf{z}_i, \mathbf{c}_1) - \mathcal{D}(\mathbf{z}_i, \mathbf{c}_2) < 0$$
$$\Rightarrow y_{i1} > y_{i2},$$

which indicates $y_{i1}^2 > y_{i2}^2 \Rightarrow y_{i1} > y_{i2}$, *i.e.*, the sample has the same cluster assignment in $\mathbf{Y}_{(t)}$ as that in the view with the scaling factor w^2.

Similarly, if $y_{i1}^1 > y_{i2}^1$ and $y_{i1}^2 < y_{i2}^2$, *i.e.*, $\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) < \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1)$ and $\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) > \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2)$, we have the same conclusions. Therefore, if we put a large scaling factor on the v-th view, the sample is more likely to be have the same cluster assignment as that in the v-th view. $\qquad\square$

Theorem 5. *(Complementarity & Noise robustness)*
Case 1: if a sample representation is informative in some views and is noisy in other views, its cluster assignment in $\mathbf{Y}_{(t)}$ is the same as that in the informative views.
Case 2: if a sample representation is noisy in all views, its cluster assignment in $\mathbf{Y}_{(t)}$ is the same as the common cluster assignments existing in these views.

Proof. **Case 1**: if a sample is informative in some views and is noisy in other views, we can treat all noisy views as one noisy view and all informative views as one informative view, *e.g.*,

$\mathbf{z}_i^1 = \begin{bmatrix} w^a \mathbf{z}_i^a & w^b \mathbf{z}_i^b \end{bmatrix}$ denotes the noisy view and $\mathbf{z}_i^2 = \begin{bmatrix} w^c \mathbf{z}_i^c & w^d \mathbf{z}_i^d \end{bmatrix}$ denotes the informative view, where $\mathbf{z}_i^a$ and $\mathbf{z}_i^b$ are noisy while $\mathbf{z}_i^c$ and $\mathbf{z}_i^d$ are informative. Letting $w^v = 1$ in Theorem 3, $\mathcal{D}(\mathbf{z}_i, \mathbf{c}_1) - \mathcal{D}(\mathbf{z}_i, \mathbf{c}_2)$ becomes

$$\mathcal{D}(\mathbf{z}_i, \mathbf{c}_1) - \mathcal{D}(\mathbf{z}_i, \mathbf{c}_2) = \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) + \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2)$$
$$= \left(\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) \right) + \left(\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) \right).$$

If $|\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1)| < \varepsilon$ and $\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) \neq \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2)$, *i.e.*, the sample is noisy in the 1-st view but informative in the 2-nd view. As ε is a sufficiently small value, we have

$$\mathcal{D}(\mathbf{z}_i, \mathbf{c}_1) - \mathcal{D}(\mathbf{z}_i, \mathbf{c}_2) \approx \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2).$$

According to the definitions, we obtain the conclusion:

$$y_{i1}^2 > y_{i2}^2 \Rightarrow \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) < \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2)$$
$$\Rightarrow \mathcal{D}(\mathbf{z}_i, \mathbf{c}_1) < \mathcal{D}(\mathbf{z}_i, \mathbf{c}_2)$$
$$\Rightarrow \mathcal{S}(\mathbf{z}_i, \mathbf{c}_1) > \mathcal{S}(\mathbf{z}_i, \mathbf{c}_2)$$
$$\Rightarrow y_{i1} > y_{i2}$$

and $y_{i1}^2 < y_{i2}^2 \Rightarrow y_{i1} < y_{i2}$, which indicates that the sample has the same cluster assignment in $\mathbf{Y}_{(t)}$ as that in $\mathbf{Y}^2$ of the informative views.

Similarly, if $\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) \neq \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1)$ and $|\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2)| < \varepsilon$, the sample has the same cluster assignment in $\mathbf{Y}_{(t)}$ as that in $\mathbf{Y}^1$ of the informative views.

Case 2: For any three clusters, $e.g.$, $\boldsymbol{\mu}_1^1$, $\boldsymbol{\mu}_2^1$, and $\boldsymbol{\mu}_3^1$ in the 1-st view, $\boldsymbol{\mu}_1^2$, $\boldsymbol{\mu}_2^2$, and $\boldsymbol{\mu}_3^2$ in the 2-nd view, if a sample is noisy in all views, there is no harm in supposing that $\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) \approx \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) < \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_3^1)$ and $\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) > \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) \approx \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_3^2)$, $i.e.$, $y_{i1}^1 \approx y_{i2}^1 > y_{i3}^1$ and $y_{i1}^2 < y_{i2}^2 \approx y_{i3}^2$. In this situation, the sample is noisy in two views and the common cluster assignments are y_{i2}^1 and y_{i2}^2 for these two views.

First, we can consider this situation $\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) \approx \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1)$ and $\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) > \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2)$. According to Theorem 3, we have

$$\mathcal{D}(\mathbf{z}_i, \mathbf{c}_1) - \mathcal{D}(\mathbf{z}_i, \mathbf{c}_2) = (w^1)^2 \left(\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_1^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) \right) + (w^2)^2 \left(\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) \right)$$
$$\approx (w^2)^2 \left(\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_1^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) \right) > 0,$$

which indicates that $y_{i1} < y_{i2}$.

Then, we can consider this situation $\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) < \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_3^1)$ and $\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) \approx \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_3^2)$. According to Theorem 3, we have

$$\mathcal{D}(\mathbf{z}_i, \mathbf{c}_2) - \mathcal{D}(\mathbf{z}_i, \mathbf{c}_3) = (w^1)^2 \left(\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_3^1) \right) + (w^2)^2 \left(\mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_2^2) - \mathcal{D}(\mathbf{z}_i^2, \boldsymbol{\mu}_3^2) \right)$$
$$\approx (w^1)^2 \left(\mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_2^1) - \mathcal{D}(\mathbf{z}_i^1, \boldsymbol{\mu}_3^1) \right) < 0,$$

which indicates that $y_{i2} > y_{i3}$.

Therefore, $y_{i1}^1 \approx y_{i2}^1 > y_{i3}^1, y_{i1}^2 < y_{i2}^2 \approx y_{i3}^2 \Rightarrow y_{i2} > y_{i1}, y_{i2} > y_{i3}$. Although the sample is noisy in the two views, the scaled representation $\mathbf{z}_i$ is informative and its cluster assignment in $\mathbf{Y}_{(t)}$ is consistent with $y_{i2}^1 \in \mathbf{Y}^1$ and $y_{i2}^2 \in \mathbf{Y}^2$. Indeed, y_{i2}^1 and y_{i2}^2 commonly denote the 2-nd cluster, and they are defined as the common cluster assignments existing in these two views. $\qquad \square$

The proofs in the above theorems can be easily extended to the situation of multiple views.

Complexity analysis. In $\mathcal{T}$-level iteration, K-means will be adopted to calculate the cluster centroids for $\mathbf{Z}_{(t)} \in \mathbb{R}^{N \times \sum_{v=1}^V d_v}$, which has the complexity of $O(NK\sum_{v=1}^V d_v)$. Eqs. (2, 6, 7, and 8) only involve the computation process with the complexity of $O(N)$. In $\mathcal{R}$-level iteration, $\min_{\mathbf{A}^v} \|\mathbf{T}\mathbf{A}^v - \mathbf{Y}^v\|_F^2$ needs the computation of Hungarian algorithm with the complexity of $O(K^3)$. The loss function of Eq. (11) of all views is with the complexity of $O(N\sum_{v=1}^V d_v + VNK)$. Both $\mathcal{T}$- and $\mathcal{R}$-level iterations can converge, and the total complexity is $O(E/T_2(T_1(NK\sum_{v=1}^V d_v) + VK^3 + VNK))$ with respect to iterations T_1 and T_2, epoch E, and data size N. The complexity to train deep autoencoders is also linear to N.

Appendix C: Setting Details and More Experimental Results

Table 2. Information of datasets

Name	V	N	K	Features
BDGP	2	2,500	5	1750/79
DIGIT	2	5,000	10	$(32 \times 32 \times 1) \times 2$
COIL	3	720	10	$(32 \times 32 \times 1) \times 3$
Amazon	3	4,790	10	$(32 \times 32 \times 3) \times 3$
NoisyBDGP	3	2,500	5	1750/79/10
NoisyDIGIT	3	5,000	10	$(32 \times 32 \times 1) \times 3$
NoisyCOIL	4	720	10	$(32 \times 32 \times 1) \times 4$
NoisyAmazon	4	4,790	10	$(32 \times 32 \times 3) \times 4$
DHA	2	483	23	110/6144
RGB-D	2	1,449	13	2048/300
Caltech	6	1,400	7	48/40/254/1984/512/928
YoutubeVideo	3	101,499	31	512/647/838

Datasets. Our experiments are conducted on 8 public datasets including 4 noise-simulated ones, as listed in Table 2.

Firstly, four normal datasets include BDGP [3], DIGIT [36], COIL [29], and Amazon [40]. Different views of these datasets often are informative views that have consistent semantic category information, and thus their hidden cluster structures are relatively easy to be discovered. Concretely, BDGP is a drosophila embryos dataset where each sample contains a visual view and a textual view. DIGIT ($i.e.$, MNIST-USPS [36]) is a dataset of handwritten arabic numbers in which each sample

has two views (paired by two digits with the same label) coming from MNIST and USPS, respectively. COIL and Amazon (*i.e.*, Multi-COIL-10 and Multi-Amazon [61]) are also two artificially constructed multi-view datasets, which are formed by following the manner of constructing MNIST-USPS [36]. COIL is a multi-view dataset and the multiple views of one object are captured by the cameras from different visual angles. Amazon consists of the commodity images, such as bicycles, bags, and earphones, where the raw samples are single-view RGB images and their augmented data are obtained by rotation, horizontal flip, and color filter to further construct the multi-view dataset.

Secondly, we construct 4 noise-simulated datasets (namely NoisyBDGP, NoisyDIGIT, NoisyCOIL, and NoisyAmazon) on the four normal datasets to test the noise robustness of methods in extreme scenarios. These datasets have both informative views and noisy views, which is helpful to understand the stability of comparison methods under extreme noise interference. For each dataset, we randomly sample noise to build an additional view and obtain NoisyBDGP, NoisyDIGIT, NoisyCOIL, and NoisyAmazon, respectively. To reduce the computational burden, all image samples are scaled to $32 \times 32 \times 1$ (DIGIT, COIL, NoisyDIGIT, and NoisyCOIL) or $32 \times 32 \times 3$ (Amazon and NoisyAmazon) pixels.

Thirdly, we also conduct experiments on four real-world datasets including DHA [21], RGB-D [72], Caltech [8], and YoutubeVideo [28]. These datasets usually have non-negligible noisy views that is from real-world multi-view environments, so that it is relatively difficult to explore their hidden cluster information. Specifically, DHA is a multimodal dataset that leverages RGB and depth features to capture human actions. RGB-D contains different indoor scenes described by image features and textual descriptions. Caltech is a popular multi-view datasets which has six views (Gabor, WM, CENTRIST, HOG, GIST, and LBP) to represent image samples. YoutubeVideo is a large-scale dataset containing video contents with cuboids histogram, HOG, and MISC features.

Implementation. For a fair comparison, we adopt the same network structures to implement the models for all deep MVC methods like previous work [11, 57, 58]. Concretely, for the vector input (BDGP, NoisyBDGP, DHA, RGB-D, Caltech, and YoutubeVideo), the autoencoder adopts the fully connected network with the architecture of $\mathbf{X}^v - 500 - 500 - 2000 - \mathbf{Z}^v - 2000 - 500 - 500 - \hat{\mathbf{X}}^v$. For the image input (DIGIT, COIL, Amazon, NoisyDIGIT, NoisyCOIL, and NoisyAmazon), the autoencoder adopts the convolutional neural network whose architecture is $\mathbf{X}^v - 32 - 64 - 64 - \mathbf{Z}^v - 64 - 32 - 32 - \hat{\mathbf{X}}^v$ with the kernel size of 4×4, the stride of 2, and the padding of 1. The activation function in networks is ReLU [9]. For the proposed MVCAN, the dimensionality of all $\{\mathbf{Z}^v\}_{v=1}^V$ is set to 10. Adam [16] with a learning rate of 0.0001 is adopted for mini-batch optimization with the batch size of 256. To make the optimization of multiple views be away from mutual interference, the deep model structure achieves the decoupling of network parameters for multiple views and we respectively construct V decoupled Adam optimizers for V different views. In $\mathcal{T}$-level iterative optimization, $T_1 = 2$. In $\mathcal{R}$-level iterative optimization, $T_2 = 100$. MVCAN is implemented by PyTorch [35], and more implementation details can be found in our code (https://github.com/SubmissionsIn/MVCAN) together with datasets and trained models. To implement the clustering and representation learning optimization objectives of MVCAN, we adopt the mean squared error (MSE) in PyTorch to compute the optimization loss of $\mathcal{L}_c^v$ and $\mathcal{L}_r^v$.

For the comparison methods, we tune the parameters for all baselines as suggested in their published work. Specially, since the original DMJC [58] conducts the single-view image clustering by exploring multiple views from the same samples, we adopt the proposed variant of DMJC-T and the multi-view data replaces the input of the single-view data in experiments. For the incomplete multi-view clustering methods (*i.e.*, DIMC-net [55], GP-MVC [49], DIMVC [60], DSIMVC [43], and CPSPAN [15]), their missing rates are set to 0. All experiments are run on a same device with NVIDIA GeForce RTX 3090 GPUs (24.0GB caches) and 11th Gen Intel(R) Core(TM) i5-11600KF @ 3.90GHz (64.0GB RAM).

More experimental results. We provide more experimental results which can not be shown in the paper due to its space limitation. The clustering effectiveness is evaluated by three widely used metrics, *i.e.*, clustering accuracy (ACC), normalized mutual information (NMI), and adjusted rand index (ARI).

Table 3. Importance of two conditions in clustering objective on four normal datasets.

	Conditions		BDGP			DIGIT			COIL			Amazon		
	Θ^v	$\mathbf{A}^v$	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
(i)	✓		0.973	0.921	0.935	0.986	0.980	0.982	0.942	0.924	0.886	0.795	0.810	0.714
(ii)		✓	0.660	0.477	0.451	0.840	0.735	0.690	0.913	0.868	0.825	0.685	0.696	0.536
(iii)	✓	✓	0.984	0.953	0.961	0.995	0.988	0.989	0.996	0.991	0.990	0.826	0.867	0.779

Table 4. Importance of two conditions in clustering objective on four noise-simulated datasets.

	Conditions		NoisyBDGP			NoisyDIGIT			NoisyCOIL			NoisyAmazon		
	Θ^v	$\mathbf{A}^v$	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
(i)	✓		0.977	0.946	0.949	0.902	0.933	0.884	0.890	0.912	0.851	0.667	0.747	0.621
(ii)		✓	0.602	0.339	0.305	0.611	0.599	0.436	0.806	0.832	0.719	0.523	0.514	0.390
(iii)	✓	✓	0.980	0.951	0.957	0.990	0.984	0.986	0.992	0.988	0.987	0.728	0.732	0.614

Table 5. Importance of two loss components in optimization on four normal datasets.

	Components		BDGP			DIGIT			COIL			Amazon		
	$\mathcal{L}_r^v$	$\mathcal{L}_c^v$	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
(a)			0.643	0.522	0.245	0.768	0.723	0.635	0.839	0.927	0.838	0.441	0.373	0.268
(b)	✓		0.948	0.842	0.874	0.787	0.747	0.658	0.826	0.888	0.787	0.498	0.452	0.322
(c)		✓	0.798	0.709	0.656	0.870	0.943	0.867	0.876	0.957	0.876	0.724	0.817	0.678
(d)	✓	✓	0.984	0.953	0.961	0.995	0.988	0.989	0.996	0.991	0.990	0.826	0.867	0.779

Table 6. Importance of two loss components in optimization on four noise-simulated datasets.

	Components		NoisyBDGP			NoisyDIGIT			NoisyCOIL			NoisyAmazon		
	$\mathcal{L}_r^v$	$\mathcal{L}_c^v$	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
(a)			0.499	0.315	0.253	0.741	0.705	0.611	0.828	0.875	0.785	0.432	0.371	0.269
(b)	✓		0.944	0.839	0.864	0.769	0.748	0.651	0.813	0.867	0.780	0.487	0.455	0.341
(c)		✓	0.726	0.574	0.381	0.591	0.702	0.504	0.686	0.782	0.612	0.423	0.423	0.258
(d)	✓	✓	0.980	0.951	0.957	0.990	0.984	0.986	0.992	0.988	0.987	0.728	0.732	0.614

(a) ACC $vs.$ T_1 (b) NMI $vs.$ T_1 (c) ARI $vs.$ T_1

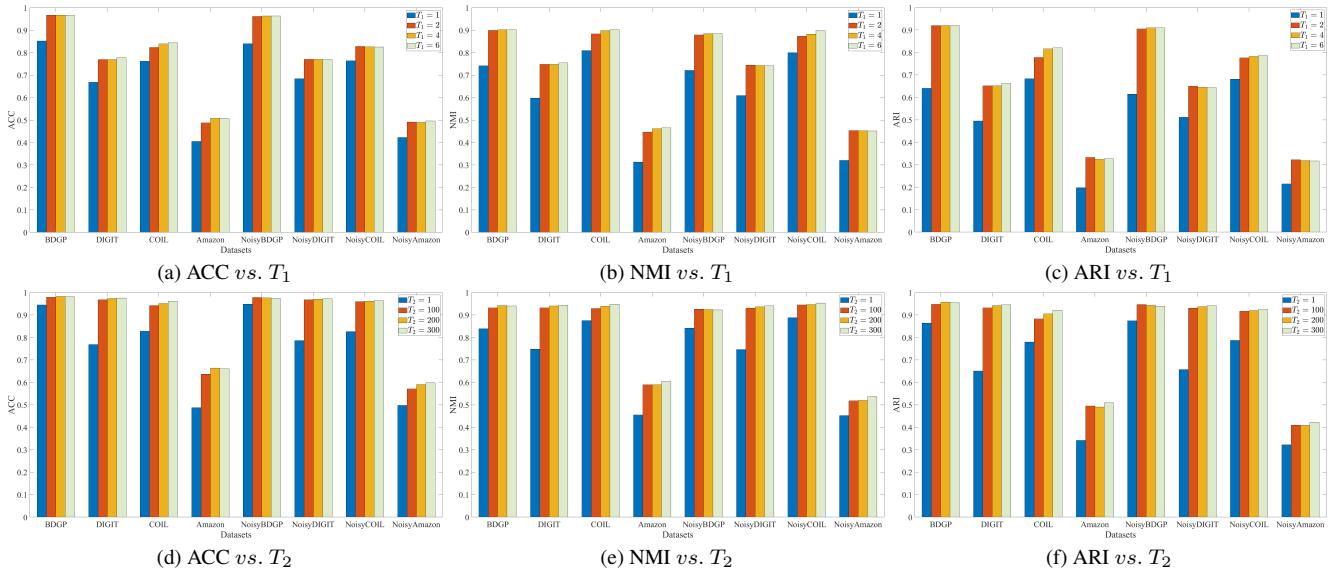(d) ACC $vs.$ T_2 (e) NMI $vs.$ T_2 (f) ARI $vs.$ T_2

Figure 3. Different training iterations in $\mathcal{T}$-level (a-c) and $\mathcal{R}$-level (d-f) of the proposed two-level multi-view iterative optimization.
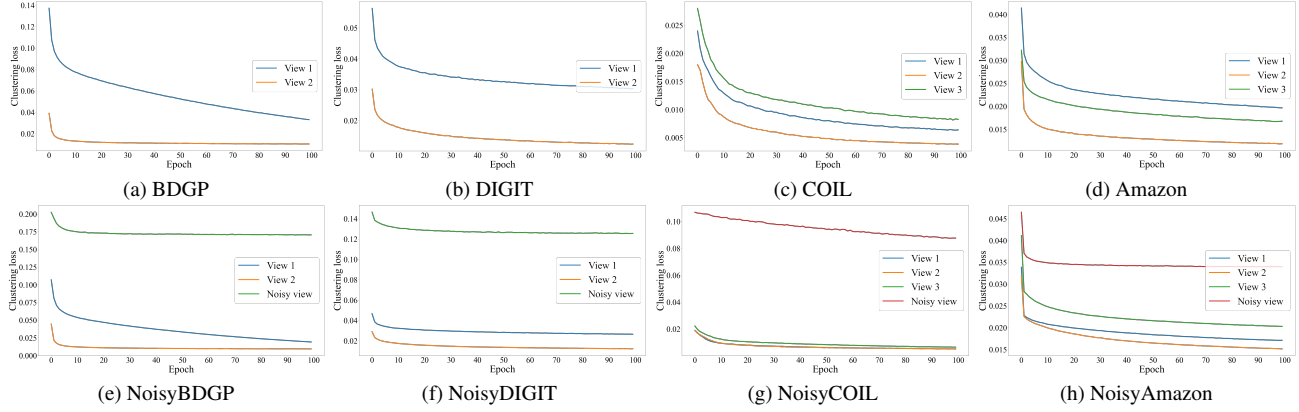
Figure 4. Loss $vs.$ Epoch on four normal datasets (a-d) and on four noise-simulated datasets (e-h).
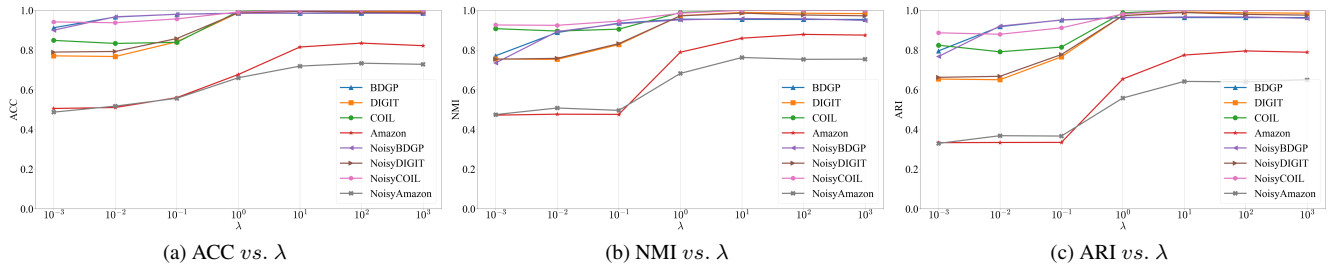

Figure 5. Clustering effectiveness $vs.$ λ on four normal datasets and on four noise-simulated datasets.
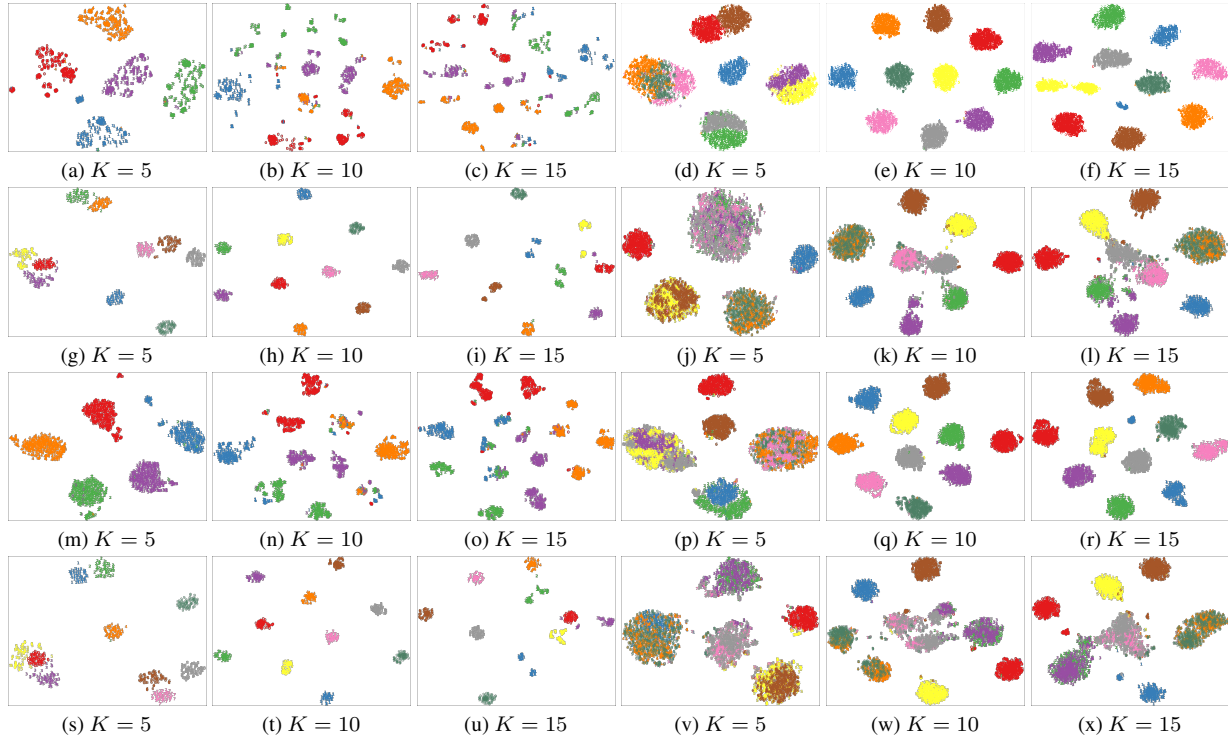

Figure 6. Visualization of the representations learned with different prior knowledge of cluster numbers on four normal and four noise-simulated datasets. BDGP (a-c), DIGIT (d-f), COIL (g-i), Amazon (j-l), NoisyBDGP (m-o), NoisyDIGIT (p-r), NoisyCOIL (s-u), NoisyAmazon (v-x).