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 [34], internet [6], and medicine [9, 26]), which can recognize the category structures and patterns without label supervision. In addition to traditional MVC [37, 41], deep learning based MVC is usually built on self-supervised methods like contrastive learning [23] and self-training [22], which has been attracting researchers' attention in recent years [4, 10, 27, 30, 45, 53] 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 [13, 30, 52], thereby outperforming single-view clustering (SVC) methods [17, 20, 25, 42]. The consistency indicates that multiple views have the consistent information which is helpful for recognizing the same category [2, 33, 49]. 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 [12, 29, 44, 47]. 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 [3, 8, 15, 46]. Some methods integrate multiple views' representations to explore their complementarity and generate a unified cluster partition for optimization as self-training manners [35, 40, 43, 45].

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

*Corresponding Author.

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 [14, 36, 38, 51, 55]. 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 [21, 30, 39, 44, 50, 54]. 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 [42]) 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:

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

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

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

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 [7, 35, 40, 43, 44]. 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 [35, 40] and late fusion [43]. They also construct the learning target $\mathbf{T}$ with Eq. (2) to train models.

Although some efforts [32, 35, 40, 48] 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:

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

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*

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

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.*, $\sum_{v\neq m^*}\|\mathbf{T} - \mathcal{F}_{\mathbf{\Theta}}(\mathbf{Y}^v|\{\mathbf{Z}^v\}_{v=1}^V)\|_F^2$. 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:

$$\begin{aligned}&\min_{\{\mathbf{\Theta}^v\}_{v=1}^V}\min_{\{\mathbf{A}^v\}_{v=1}^V}\sum_{v=1}^{V}\|\mathbf{TA}^v - \mathcal{F}^v_{\mathbf{\Theta}^v}(\mathbf{Y}^v|\mathbf{Z}^v)\|_F^2\\ &s.t.\ \mathbf{\Theta}^a \cap \mathbf{\Theta}^b = \varnothing, a, b \in \{1,2,\ldots,V\}, a \neq b,\\ &\quad \mathbf{A}^v(\mathbf{A}^v)^T = \mathbf{I}_K, \mathbf{A}^v \in \{0,1\}^{K\times K},\end{aligned} \tag{5}$$

where we write $\mathbf{Y}^v = \mathcal{F}^v_{\mathbf{\Theta}^v}(\mathbf{Y}^v|\mathbf{Z}^v)$ whose calculation follows Eq. (1). $\mathbf{Z}^v = \mathcal{E}^v_{\mathbf{\Phi}^v}(\mathbf{X}^v)$ and $\mathcal{E}^v_{\mathbf{\Phi}^v}$ 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 $\{\boldsymbol{\mu}_j^v\}_{j=1}^K$ and $\mathbf{\Phi}^v$ of the v-th view. We leverage $\mathbf{\Theta}^a \cap \mathbf{\Theta}^b = \varnothing, a, b \in \{1,2,\ldots,V\}, a \neq b$ 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, $\min_{\{\mathbf{A}^v\}_{v=1}^V}\sum_{v=1}^V\|\mathbf{TA}^v - \mathcal{F}^v_{\mathbf{\Theta}^v}(\mathbf{Y}^v|\mathbf{Z}^v)\|_F^2$, which leads to $\sum_{v=1}^V\|\mathbf{TA}^v - \mathcal{F}^v_{\mathbf{\Theta}^v}(\mathbf{Y}^v|\mathbf{Z}^v)\|_F^2 \le \sum_{v=1}^V\|\mathbf{T} - \mathcal{F}^v_{\mathbf{\Theta}^v}(\mathbf{Y}^v|\mathbf{Z}^v)\|_F^2$. 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 $\left[\mathbf{Z}^1 \quad \mathbf{Z}^2 \quad \dots \quad \mathbf{Z}^V\right] \in \mathbb{R}^{N\times\sum_v d_v}$ and the scaling matrix $\mathbf{W}_{(t)} \in \mathbb{R}^{\sum_v d_v\times\sum_v d_v}$:

$$\begin{aligned}\mathbf{Z}_{(t)} &= \mathcal{H}(\mathbf{Z}_{(t)}|\mathbf{W}_{(t)}, \mathbf{Z}^1, \mathbf{Z}^1, \dots, \mathbf{Z}^V) \\ &= \left[\mathbf{Z}^1 \quad \mathbf{Z}^2 \quad \dots \quad \mathbf{Z}^V\right]\mathbf{W}_{(t)} \\ &= \left[\mathbf{Z}^1 \quad \mathbf{Z}^2 \quad \dots \quad \mathbf{Z}^V\right]\begin{bmatrix} w^1_{(t)}\mathbf{I}^1 & & & \\ & w^2_{(t)}\mathbf{I}^2 & & \\ & & \ddots & \\ & & & w^V_{(t)}\mathbf{I}^V \end{bmatrix},\end{aligned} \tag{6}$$

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 $w^v_{(t)} \in \mathbb{R}$ 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:

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

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 [17] 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 $w^v_{(t+1)}$ for each view by

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

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 $\hat{y}^v_i = \arg\max_j y^v_{ij}$, $\hat{y}_{i(t)} = \arg\max_j y_{ij(t)}$, 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 $\mathbf{A}^{v*} = \min_{\mathbf{A}^v} \|\mathbf{T}\mathbf{A}^v - \mathbf{Y}^v\|_F^2$ with Hungarian algorithm. For each view, $\mathbf{A}^{v*}$ produces a different learning target $\mathbf{T}^v = \mathbf{T}\mathbf{A}^{v*}$ and then Eq. (5) can be transformed into the following clustering objective (denoted by $\mathcal{L}_c^v$):

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

Additionally, we follow previous deep MVC methods [31, 35, 40, 43, 45] and adopt deep autoencoders (a popular self-supervised representation learning method) to learn the new representations of multi-view data. Letting $\mathcal{E}^v_{\mathbf{\Phi}^v}$ and $\mathcal{D}^v_{\mathbf{\Psi}^v}$ 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 $\hat{\mathbf{X}}^v = \mathcal{D}^v_{\mathbf{\Psi}^v}(\mathbf{Z}^v)$ is only related to $\mathbf{Z}^v = \mathcal{E}^v_{\mathbf{\Phi}^v}(\mathbf{X}^v)$, and the representation learning objective (denoted by $\mathcal{L}_r^v$) is:

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

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

$$\mathcal{L}^v = \mathcal{L}_r^v + \lambda\mathcal{L}_c^v, \tag{11}$$

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 MVCAN works with the following definitions and theorems:

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 $\{\{\boldsymbol{\mu}_j^v\}_{j=1}^K\}_{v=1}^V$ 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 $\{\boldsymbol{\mu}_j^v\}_{j=1}^K$ 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 $\{\mathbf{\Phi}^v, \mathbf{\Psi}^v, \{\boldsymbol{\mu}_j^v\}_{j=1}^K, \mathbf{A}^v\}_{v=1}^V$

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 $\mathbf{b}$,*

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

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, $\mathbf{z}$, $\boldsymbol{\mu}$ - 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.*

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 [1], DIGIT [24], COIL [19], and Amazon [28]. 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 [42]	92.6	81.9	80.9	78.9	76.6	81.5	47.0	32.5
DEC-WorstV [42]	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 [43]	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 [40]	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 [35]	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 [32]	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 [44]	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 [30]	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 [31]	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 [8]	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 [45]	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 [42]	72.6	79.3	43.6	40.1	88.2	81.6	20.9	20.4
DEC-WorstV [42]	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 [43]	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 [40]	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 [35]	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 [32]	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 [44]	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 [30]	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 [31]	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 [8]	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 [45]	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 [11], RGB-D [54], Caltech [5], and YoutubeVideo [18].

Comparison methods. We compare our MVCAN with the following 10 self-supervised clustering algorithms. To be specific, DEC [42] is a popular deep SVC method and we leverage this baseline to investigate the side effects of NVD on MVC methods. DMJC [43], DIMC-net [40], GP-MVC [35], DIMVC [44], and SDMVC [45] are DEC-based deep MVC methods which usually establish consistent soft labels for achieving clustering consistency. DMJC [43], DIMC-net [40], GP-MVC [35], and DSMVC [30] mainly incorporate weighting strategies to obtain fused representations. CoMVC [32], DSIMVC [31], and CPSPAN [8] 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 [42]	92.6	81.9	80.9	78.9	76.6	81.5	47.0	32.5
DEC-WorstV [42]	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 [43]	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 [40]	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 [35]	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 [32]	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 [44]	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 [30]	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 [31]	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 [8]	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 [45]	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 [32, 48], 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.

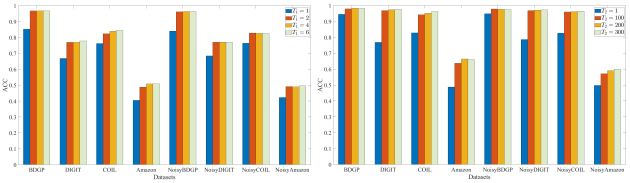
Table 4. Importance of two conditions in clustering objective.

	Conditions		BDGP		DIGIT		NoisyBDGP		NoisyDIGIT	
	$\mathbf{\Theta}^v$	$\mathbf{A}^v$	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 in Table 4, (i) $\mathbf{\Theta}^v$ 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 $\{\mathbf{\Theta}^v\}_{v=1}^V$ 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.



(a) ACC $vs.$ T_1 (b) ACC $vs.$ T_2

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

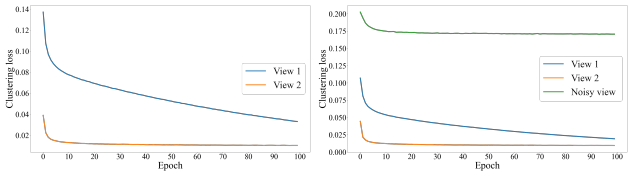
Two loss components in optimization. Table 5 lists the results of MVCAN with different loss components, where (a) denotes the clustering results of K-means on the direct concatenation of multi-view data. Compared with (a), both (b) and (c) can obtain improvements due to the representation learning objective 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 $\mathbf{T}$. 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 $T_2 = 100$.

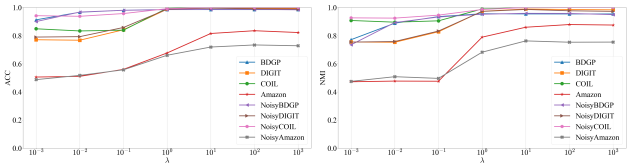
4.4. Model Analysis

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.

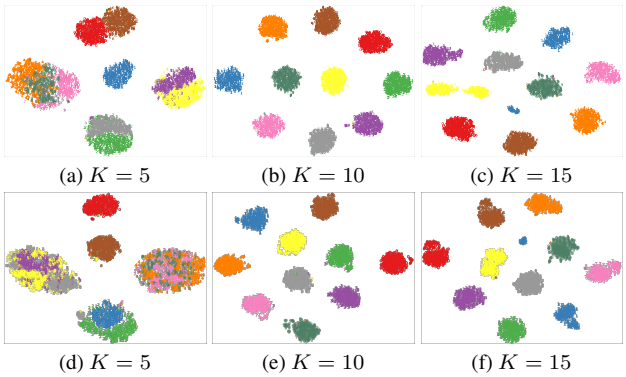


(a) ACC vs. λ (b) NMI vs. λ

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 MVCAN 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 [16] 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, MVCAN 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.

Acknowledgment

This work was supported in part by the National Key Research & Development Program of China under Grant 2022YFA1004100, in part by the Medico-Engineering Cooperation Funds from University of Electronic Science and Technology of China under Grant ZYGX2022YGRH009 and Grant ZYGX2022YGRH014.

References

[1] 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

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

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

[4] 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

[5] 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

[6] 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

[7] 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

[8] 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

[9] 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

[10] 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

[11] 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

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

[13] 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

[14] 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

[15] 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

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

[17] 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

[18] 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

[19] 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

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

[21] 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

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

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

[24] 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

[25] 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

[26] 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

[27] 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

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

[29] 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

[30] 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

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

[32] 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

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

[34] 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

[35] 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

[36] 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

[37] 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

[38] 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

[39] 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

[40] 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

[41] 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

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

[43] 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

[44] 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

[45] 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

[46] 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

[47] 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

[48] 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

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

[50] 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

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

[52] 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

[53] 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

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

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