Partial Multi-View Clustering via Self-Supervised Network

Wei Feng[1], Guoshuai Sheng[2], Qianqian Wang[2*], Quanxue Gao[2], Zhiqiang Tao[3], Bo Dong[4]

[1] School of Computer Science and Technology, Xi'an Jiaotong University, Xi'an, Shaanxi, China, 710049
[2] School of Telecommunications Engineering, Xidian University, Xi'an, Shaanxi, China, 710071
[3] School of Information, Rochester Institute of Technology, Rochester, NY, USA, 14623
[4] School of Continuing Education, Xi'an Jiaotong University, Xi'an, Shaanxi, China, 710049
weifeng.ft@xjtu.edu.cn, agussheng@foxmail.com, qqwang@xidian.edu.cn, qxgao@xidian.edu.cn, zhiqiang.tao@rit.edu, dong.bo@xjtu.edu.cn

Abstract

Partial multi-view clustering is a challenging and practical research problem for data analysis in real-world applications, due to the potential data missing issue in different views. However, most existing methods have not fully explored the correlation information among various incomplete views. In addition, these existing clustering methods always ignore discovering discriminative features inside the data itself in this unsupervised task. To tackle these challenges, we propose **P**artial Multi-**V**iew **C**lustering via **S**elf-**S**upervised **N**etwork (PVC-SSN) in this paper. Specifically, we employ contrastive learning to obtain a more discriminative and consistent subspace representation, which is guided by a self-supervised module. Self-supervised learning can exploit effective cluster information through the data itself to guide the learning process of clustering tasks. Thus, it can pull together embedding features from the same cluster and push apart these from different clusters. Extensive experiments on several benchmark datasets show that the proposed PVC-SCN method outperforms several state-of-the-art clustering methods.

Introduction

Multi-view data contain multiple features from different views, such as sensors, modalities, viewpoints, sources, etc. Nowadays, multi-view data become quite ubiquitous in practice due to abundant multi-media data collection equipment. For example, a video has image view, text view, and audio view (Wang et al. 2023a). However, it is difficult to process multi-view data because of the lack of reliable labels. A basic solution is to employ multi-view clustering (MVC) (Li et al. 2019; Wen et al. 2022; Xu et al. 2022) to separate the unlabeled multi-view data to different groups where data in the same groups may belong to the same class with high probability. Owing to its promising performance, MVC has been well researched, and numerous MVC methods (Tao et al. 2020; Zhang et al. 2017; Jiang et al. 2022) are proposed. Traditional MVC methods are based on subspace learning, graph learning, spectral learning, and non-negative matrix factorization. For example, (Gao et al. 2015) proposes multi-view subspace clustering (MVSC) which performs clustering on a consistent structure learned from the subspace of each view; (Wang et al. 2018b) develops multi-view spectral clustering by utilizing low-rank matrix factorization. Though they are effective in multi-view data, they are unable to capture complex features of multi-view data. To overcome the weakness and inspired by deep learning, deep MVC methods (Chen et al. 2023; Wang et al. 2020a) are explored to capture non-linear relationships of multi-view data. For example, (Li et al. 2019) designed deep MVC via adversarial networks; (Andrew et al. 2013) developed deep canonical correlation analysis (CCA) method for cross-view data (Gao et al. 2020).

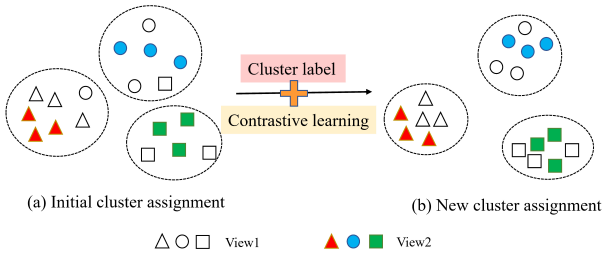

Figure 1: The illustration of changes in cluster assignments, where solid points and hollow points with the same shape represent different views, and different shapes represent different clusters.

Although MVC methods have achieved successful results, they cannot work well on the multi-view data, of which some samples lack information in one or two views, i.e., partial/incomplete multi-view data (Xu, Tao, and Xu 2015; Wang et al. 2021; Zhang et al. 2020). This makes these methods difficult to apply in real-world applications since partial multi-view data is inevitable in practice because of noises, sensor failure, and transmission loss. To tackle this problem, several partial multi-view clustering (PMVC)/incomplete multi-view clustering (IMC) methods (Liu et al. 2018; Shao et al. 2016) are proposed. Original PMVC methods are mainly designed for two-view data (Zhao, Liu, and Fu 2016; Wang et al. 2018a, 2023b). For example, (Li, Jiang, and Zhou 2014) uses NMF and ℓ-norm to obtain a complete subspace for partial two-view clustering, named PVC. To well process partial multi-view data with more than two views, (Shao, He, and Philip 2015) develops multiple incomplete

*Corresponding author.

views clustering (MIC) via weighted NMF with $\ell_{2,1}$-norm regularization; (Wen, Xu, and Liu 2018) proposed incomplete multi-view spectral clustering by adaptive graph learning (IMSC). These methods have made great progress in partial multi-view clustering. To further excavate complex features of partial multi-view data with deep learning, (Wang et al. 2018a) employs generative adversarial network (GAN) to integrate clustering with missing sample completion and further alleviate performance degradation due to data missing.

However, these PMVC methods still have two limitations. First, existing methods cannot learn an accurate common clustering structure from partial multi-view data with large difference among each view. Thus, how to further explore the hidden correlation of each view with large difference still lacks exploration. Second, since clustering is an unsupervised method, existing PMVC methods ignore the discriminative information in samples. Though some MVC methods consider using self-supervised method (Sun et al. 2019) to improve clustering performance, there is little research on partial multi-view clustering. As a result, existing PMVC methods cannot capture enough category information for better clustering performance.

To overcome these limitations above, in this paper, we propose a novel Partial Multi-view Clustering via Self-Supervised Network (PVC-SSN). More specifically, we utilize a deep multi-view contrastive encoder network to reduce the difference of latent subspace by maximizing the consistency among multiple views. Among this latent subspace, the self-expression layer is embedded to learn a consistent subspace representation. Moreover, multi-view decoder network is designed to obtain reconstructed samples, which can ensure the validity of subspace representation. Fig. 1 illustrates the contributions of our model. In summary, the contributions of our PVC-SSN method are as follows:

- We propose a partial multi-view clustering via self-supervised network (PVC-SSN), which could maximize the feature consistency of partial multi-view data, and further learn a more discriminative subspace representation by introducing contrastive learning.
- We design a self-supervised module to guide the selection of positive/negative samples for contrastive learning. Furthermore, we construct a fusion mechanism that can adaptively weight representations from different views and finally obtain a discriminative common representation.
- Extensive experiments on several benchmark datasets show that the proposed PVC-SSN method outperforms several state-of-the-art clustering methods.

Methodology

Notations

Given a set of unlabeled complete multi-view data $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, ..., \mathbf{X}^v\}$, $\mathbf{X}^i \in \mathbf{R}^{d_i \times n}$, where n and d_i are respectively the total number of samples and the feature dimension corresponding to each view. For all samples, we randomly remove a certain percentage of samples and divide the remaining data into two parts: paired data(samples are available for all views) $\mathbf{X}_p^i = (\boldsymbol{x}_1^i, \boldsymbol{x}_2^i, ..., \boldsymbol{x}_p^i) \in \mathbf{R}^{d_i \times p}, i = 1, 2, ..., v$ and unpaired data (samples are available for one view only) $\mathbf{X}_u^i = (\boldsymbol{x}_{p+1}^i, \boldsymbol{x}_{p+2}^i, ..., \boldsymbol{x}_{p+u}^i) \in \mathbf{R}^{d_i \times u}, i = 1, 2, ..., v$, where p and u are samples of paired and unpaired data. For each view, we construct a view encoder $\boldsymbol{E}_i, i = 1, 2, ..., v$, a corresponding contrastive head $\boldsymbol{G}_i, i = 1, 2, ..., v$, and a corresponding view decoder $\boldsymbol{D}_i, i = 1, 2, ..., v$.

Network Architecture

Fig.2 illustrates the overall pipeline of our method, which consists of three key components: (a) multi-view contrastive encoder network, (b) self-expression learning layer, (c) multi-view decoder network. First, we send partial multi-view data $[\mathbf{X}_p^i, \mathbf{X}_u^i]$ to multi-view contrastive encoder network and obtain latent subspace features $[\mathbf{Z}_p^i, \mathbf{Z}_u^i]$. Since it is crucial to learn a better similarity for multi-view data in the subspace clustering task, we thus maximize the consistency of latent representations from different views via multi-view contrastive loss, which makes representations more discriminative and cluster-friendly. In the latent space, we perform a feature fusion operation to eliminate redundant information and obtain a stronger feature representation $\mathbf{Z}$. To obtain a shared subspace representation, we then perform self-expressive property on the feature $\mathbf{Z}$. To make the feature obtain a remarkable clustering structure, we use the pseudo-labels $\mathbf{P}$ obtained from the self-expression coefficient matrix $\mathbf{S}$ to guide the construction of positive/negative pairs for contrastive learning. The label information can effectively guide the training process to learn a better cluster representation. Finally, we add a multi-view decoder network to ensure the validity of subspace representation.

Multi-view Contrastive Encoder Network We design a multi-view contrastive encoder network to transform multi-view data into low-dimensional latent space. Given partial data $\mathbf{X}^i = [\mathbf{X}_p^i, \mathbf{X}_u^i] \in \mathbf{R}^{d_i \times (p+u)}, i = 1, 2, ..., v$, we construct a multi-view encoder network $\boldsymbol{E}_i$ and transform i-th-view data into low-dimensional subspace by the mapping function $\mathbf{Z}^i = f(\mathbf{X}^i, \theta) = [\mathbf{Z}_p^i, \mathbf{Z}_u^i] \in \mathbf{R}^{c \times (p+u)}$ is i-th-view subspace feature and θ is the network parameter matrix. After obtaining the low-dimensional feature representation of each single-view data, we do not directly devise a contrastive work to contrast paired features $\mathbf{Z}_p^i$ of multiple views. Instead, we design a one-layer nonlinear contrastive head $\boldsymbol{G}_i$ to map the paired features to contrastive embedding features via $\mathbf{Q}_p^i = g_i(\mathbf{Z}_p^i) \in \mathbf{R}^{k \times p}$, where k is the number of clusters. Furthermore, we normalize the output of the contrastive head and perform supervised contrastive learning in the contrastive embedding space.

Self-expression Learning Layer The latent subspace features $\mathbf{Z}^i = [\mathbf{Z}_p^i, \mathbf{Z}_u^i] \in \mathbf{R}^{c \times (p+u)}, i = 1, 2, ..., v$ can be obtained from multi-view contrastive encoder network, which the paired data $\{\mathbf{Z}_p^1, \mathbf{Z}_p^2, ..., \mathbf{Z}_p^v\}$ are fused into a common subspace representation $\mathbf{Z}_p$ in the weighted fusion manner. In the process of fusion, we first initialize the same weight parameters as $1/v$ for each view and adaptively learn the weight of each view during the training process.

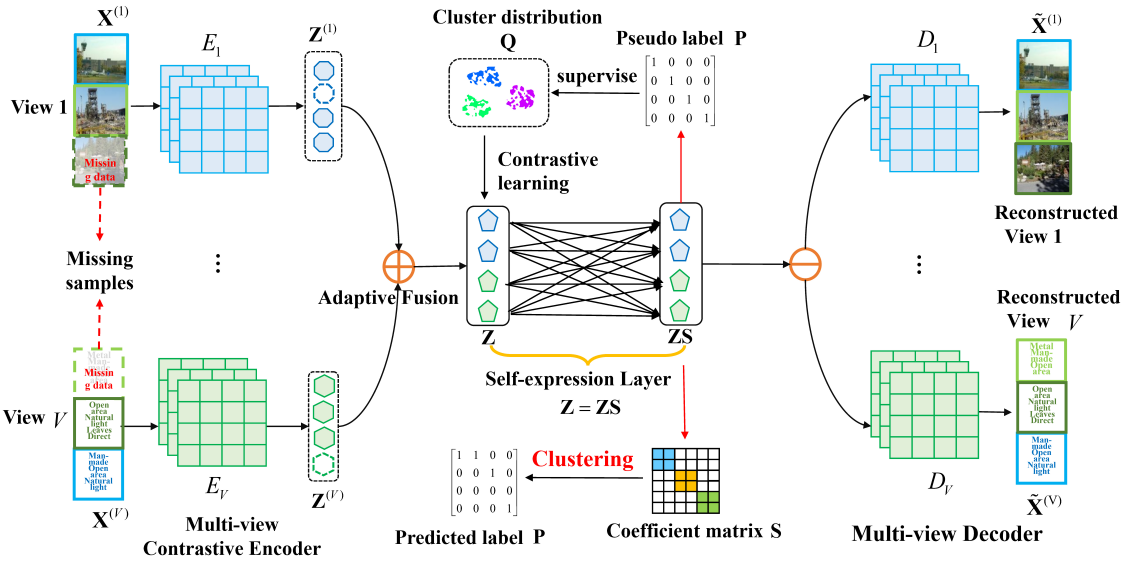

Figure 2: The framework of our proposed model PVC-SSN. It consists of a multi-view contrastive encoder, a self-expression layer, and a multi-view decoder. The multi-view contrastive encoder is responsible for encoding the original high-dimensional samples into low-dimensional subspace features; the self-expression network is responsible for learning a sample correlation coefficient matrix for spectral clustering; the multi-view decoder restores the subspace features to the original input data.

Afterwards, we fuse all views by $\sum_{i=1}^{v} \alpha_i \mathbf{Z}_p^i / \sum_{i=1}^{v} \alpha_i$. We finally splice the unpaired data $\{\mathbf{Z}_u^1, \mathbf{Z}_u^2, ..., \mathbf{Z}_u^v\}$ with the common representation $\mathbf{Z}_p$ to obtain an overall subspace feature $\mathbf{Z} = [\mathbf{Z}_p, \mathbf{Z}_u^1, ..., \mathbf{Z}_u^v] \in \mathbf{R}^{c \times n}$. The self-expression learning is used to learn a subspace distributed representation *i.e.*, the self-expression coefficient matrix and can be considered as the matrix multiply, *i.e.*, $\mathbf{Z} = \mathbf{ZS}$, where $\mathbf{S} \in \mathbf{R}^{n \times n}$ is the self-expression coefficient matrix and it has the property of being distributed in blocks. Then we can leverage the matrix $\mathbf{S}$ to construct the affinity matrix $\mathbf{C}$, *i.e.*, $\mathbf{C} = \frac{1}{2}(|\mathbf{S}| + |\mathbf{S}|^\top)$. Finally, we conduct spectral clustering on the affinity matrix $\mathbf{C}$ to obtain clustering result $\mathbf{P}$, which will be fed back to the multi-view contrastive encoder network to guide the learning of common representation.

Multi-view Decoder Network We employ the decoding networks to reconstruct the input data to ensure that the potential subspace features that are learned from the encoding networks can well reflect the structural characteristics of the original data. More specifically, the representation $\mathbf{ZS}$ will be divided to multiple representations $[(\mathbf{ZS})_p^i, (\mathbf{ZS})_u^i]$, $i = 1, 2, ..., v$, which are sent to the corresponding decoder network and we will obtain reconstruction data $\widetilde{\mathbf{X}}^i = f_d([(\mathbf{ZS})_p^i, (\mathbf{ZS})_u^i])$.

Objective Function

According to the proposed framework, the objective function of the model consists of three parts: multi-view contrastive loss, self-expression loss, and reconstruction loss. The following will introduce these loss functions in turn.

Multi-view Contrastive Loss For our encoders to better learn discriminative information of different categories, supervised contrastive learning is applied into our encoder network to maximize the agreement of positive pairs and to minimize the agreement of negative pairs. Previous work has used data augmentation to obtain positive and negative pairs in the latent space, and then update the network via a self-supervised contrastive loss. In this work, we use the clustering results obtained in the last iteration to construct positive and negative sample pairs, so that samples of the same cluster are positive samples, and samples of different clusters are negative samples. Given a sample x_a^i of the object a from the i-th view paired data $\mathbf{X}_p^i$, we can obtain its corresponding representation z_a^i and contrastive embedding feature q_a^i via the i-th view contrastive head map $q_a^i = g_i(z_a^i)$. Then we can obtain the i-th view contrastive embedding features $\mathbf{Q}_p^i = (q_1^i, q_2^i, ..., q_p^i)$ and construct a multi-view contrastive embedding space $\mathbf{Q} = (\mathbf{Q}_p^1, \mathbf{Q}_p^2, ..., \mathbf{Q}_p^v) \in \mathbf{R}^{k \times pv}$. We construct positive pair with q_a^i and other feature points of the same class as q_a^i in embedding space. Specifically, we define the set of positive samples of q_a^i as $B(q_a^i) \equiv \{q_b^j \in \mathbf{Q} : q_b^j \neq q_a^i, y_b = y_a\}$, where y_a and y_b represent the label of the object a and b. The set of positive and negative samples of q_a^i is defined as $C(q_a^i) \equiv \{q_c^k \in \mathbf{Q} : q_c^k \neq q_a^i\}$. Therefore, $C(q_a^i)$ minus $B(q_a^i)$ is equal to the set of negative samples. Finally, we get our multi-view contrastive loss as follows

$$L_{mcl} = \sum_{i=1}^{v} \sum_{a=1}^{p} \frac{-1}{|B(q_a^i)|} \sum_{B(q_a^i)} \log \frac{\exp\left(q_a^i \cdot q_b^j / \tau\right)}{\sum_{C(q_a^i)} \exp\left(q_a^i \cdot q_c^k / \tau\right)}, \tag{1}$$

where $|B(q_a^i)|$ is the number of positive samples of q_a^i and τ denotes a temperature parameter to control the degree of attention to hard negative pairs. In our experiments, we let $\tau = 0.1$. Note that each feature point in the embedding space has been normalized.

Self-expression Loss Through the self-expression learning layer, the self-expression coefficient matrix $\mathbf{S}$ can be learned, and subspace feature $\mathbf{Z}$ is estimated, *i.e.*, $\mathbf{Z} = \mathbf{ZS}$. The self-expression learning function is expressed as follows

$$L_{se} = \frac{1}{2}||\mathbf{Z} - \mathbf{ZS}||_{\mathrm{F}}^2 + ||\mathbf{S}||_{\mathrm{F}}^2 \quad s.t. \quad diag(\mathbf{S}) = 0, \quad (2)$$

where $||\cdot||_{\mathrm{F}}^2$ is the square of the Frobenius norm, and $diag(\cdot)$ refers to extracting the diagonal elements of the matrix. The constraint $diag(\mathbf{S}) = 0$ aims to prevent the trivial solutions of $\mathbf{S} = \mathbf{I}$.

Reconstruction Loss To ensure the validity of subspace representation, we use the mean square error between input original data $\mathbf{X}^i$ and reconstructed data $\tilde{\mathbf{X}}^i$ as reconstruction loss of the v-th view. The above reconstruction loss is defined as

$$L_{re} = \sum_{i=1}^{v} ||\mathbf{X}^i - \tilde{\mathbf{X}}^i||_{\mathrm{F}}^2, \quad (3)$$

Overall Objective Combining Eqs. (1) and (2) can get the loss of pre-training stage as follows

$$L_{\mathrm{pre}} = L_{re} + L_{se}, \quad (4)$$

In the normal training stage, by integrating reconstruction loss, self-expression loss and multi-view contrastive loss, we have the following objective function of PVC-SSN

$$L = L_{re} + \lambda_1 L_{se} + \lambda_2 L_{mcl}, \quad (5)$$

where λ_1 and λ_2 are trade-off parameters to adjust the impact of each term in all objective functions. L_{re}, L_{se}, and L_{mcl} represent the reconstruction loss, the self-expression learning loss and the multi-view contrastive loss for our PVC-SSN method, respectively.

Algorithm 1: PVC-SSN

Input: Partial multi-view data: $\mathbf{X}^i = [\mathbf{X}_p^i, \mathbf{X}_u^i] \in \mathbf{R}^{d_i \times (p+u)}, i = 1, 2, ..., v$; Learning rate: 0.0001.
while pre-training not converge **do**
(1) Pre-train the networks using Eq. (4);
(2) Optimize network parameters $\theta_{e_1}, \theta_{e_2}$ of encoders and $\theta_{d_1}, \theta_{d_2}$ of decoders;
(3) Save the cluster assignments as initial pseudo-labels.
end pre-training
while training not converge **do:**;
(3) Formally train the network using Eq. (5);
(4) Adjust trade-off parameters: λ_1, λ_2, λ_3;
(5) Update network parameters: $\theta_i, i = 1, 2, ..., v$;
(6) Update the self-expression coefficient matrix: $\mathbf{S}$.
end training
(7) **return** the self-expression coefficient matrix $\mathbf{S}$;
(8) Compute the affinity matrix $\mathbf{C} = \frac{1}{2}(|\mathbf{S}| + |\mathbf{S}|^\top)$;
(9) Perform spectral clustering on the affinity matrix $\mathbf{C}$.

Experiments

Experimental Settings

Databases To evaluate the effectiveness of the proposed PVC-SSN method, we conduct experiments on three benchmark datasets, *i.e,* BDGP (Cai et al. 2012), MNIST (LeCun 1998) and HW (van Breukelen et al. 1998).

- **BDGP dataset**: This dataset has a total of 2500 samples, which are divided into 5 categories. The image and text modalities of each sample are represented by 1×1750 and 1×79 vectors, respectively.
- **MNIST dataset**: MNIST collects 70000 handwritten digits from 0 to 9, which are divided into training set and testing set. In our experiment, only the training set of 4000 handwritten digits is used. We extract its original image, edge, LBP and encoder features for our experimental four views.
- **HW dataset**: The dataset collects 2000 digits from 0 to 9, in which each class has 200 samples with 6 kinds of features. These features include 216 profile correlations (FAC), 76 Fourier coefficients for two-dimensional shape descriptors (FOU), 64 Karhunen-Loeve coefficients (KAR), 240 pixel feature (PIX), 47 rotational invariant Zernike moment (ZER), 6 morphological features (MOR). In our experiment, only the first three view of HW dataset is used.

Comparison Methods In the experiments, we compare the proposed PVC-SSN model with several state-of-the-art methods including: (1) one single-view clustering method: Spectral Clustering (SC) (Ng, Jordan, and Weiss 2002); (2) three multi-view clustering methods: Auto-weighted Multiple Graph Learning (AMGL) (Nie et al. 2016), Robust Multi-view Spectral Clustering (RMSC) (Xia et al. 2014), Feature Concatenation Spectral Clustering (ConSC) (Kumar, Rai, and Daume 2011); (3) four incomplete data clustering methods: Incomplete Multi-Modal Visual Data Grouping (IMG) (Zhao, Liu, and Fu 2016), Multi-View Clustering using Graph Regularized NMF (GPVC) (Rai et al. 2016), Generative Partial Multi-View Clustering (PVC-GAN) (Wang et al. 2018a), and Incomplete Cross-Modal Subspace Clustering (iCmSC) (Wang et al. 2020b), and several classical clustering methods.

Evaluation Metric We measure the performance of comparison methods and our proposed method using the cluster accuracy (ACC) (Cai, He, and Han 2005) and the normalized mutual information(NMI) (Estévez et al. 2009). For partial multi-view data clustering task, we set five groups of missing ratios as $(0.1, 0.3, 0.5, 0.7, 0.9)$ for each dataset in our experiments. Under the same missing ratio, we randomly miss 10 times for complete data and run them to average the ACC and NMI values. The data with high similarity should be clustered into the same group, and different data into different groups. Therefore, the bigger the values of ACC and

Methods \ Missing rate	0.9	0.7	0.5	0.3	0.1
SC1	0.3296±0.0054	0.3539±0.0054	0.3845±0.0067	0.4103±0.0042	0.4404±0.0039
SC2	0.4748±0.0131	0.5169±0.0174	0.5692±0.0159	0.6139±0.0121	0.6716±0.0136
AMGL	0.2524±0.0349	0.2357±0.0180	0.2538±0.0155	0.2807±0.0125	0.2958±0.01952
RMSC	0.3395±0.0050	0.3683±0.0051	0.3907±0.0045	0.4233±0.0048	0.4499±0.0022
ConSC	0.2781±0.0411	0.2230±0.0148	0.2139±0.0078	0.2106±0.0058	0.2884±0.0896
GPVC	0.5015±0.0438	0.5424±0.0537	0.6277±0.0402	0.6833±0.0931	0.7546±0.1091
IMG	0.4373±0.0100	0.4508±0.0254	0.4868±0.0147	0.5055±0.0131	0.5176±0.0415
PVC-GAN	0.5210±0.0090	0.6711±0.0107	0.8631±0.0043	0.9154±0.0107	0.9498±0.0026
iCmSC	0.5901±0.0079	0.7477±0.0043	0.8845±0.0030	0.9210±0.0013	0.9569±0.0031
PVC-SSN	**0.6032±0.0144**	**0.8036±0.0109**	**0.9156±0.0087**	**0.9304±0.0073**	**0.9616±0.0035**

Table 1: The clustering accuracy rate(ACC)(%) on BDGP dataset for two views.

Methods \ Missing rate	0.9	0.7	0.5	0.3	0.1
SC1	0.4398±0.0140	0.4665±0.0098	0.4731±0.0202	0.5070±0.0355	0.5430±0.0220
SC2	0.3324±0.0147	0.3366±0.0172	0.3532±0.0149	0.3696±0.0074	0.3769±0.0120
SC3	0.4159±0.0193	0.4429±0.0086	0.4811±0.0129	0.4901±0.0129	0.5083±0.0195
SC4	0.3088±0.0068	0.3186±0.0115	0.3310±0.0154	0.3522±0.0102	0.3791±0.0142
AMGL	0.1558±0.0155	0.1412±0.0218	0.1524±0.0343	0.2415±0.0631	0.3346±0.0288
RMSC	0.3492±0.0077	0.4150±0.0294	0.4575±0.0233	0.4960±0.0174	0.5144±0.0204
ConSC	0.3704±0.0275	0.3581±0.02318	0.3674±0.0131	0.4137±0.0396	0.5088±0.0299
GPVC	0.3525±0.0238	0.3864±0.0104	0.4238±0.0446	0.4401±0.0150	0.4644±0.0423
IMG	0.4655±0.0186	0.4640±0.0213	0.4613±0.0146	0.4592±0.0146	0.4622±0.0151
PVC-GAN	0.4517±0.0086	0.4836±0.0071	0.5280±0.0078	0.5202±0.0070	0.5340±0.0073
iCmSC	0.5089±0.0074	0.5665±0.0163	0.5834±0.0089	0.6012±0.0068	0.6319±0.0097
PVC-SSN	**0.5775±0.0094**	**0.5835±0.0073**	**0.6277±0.0104**	**0.6335±0.0098**	**0.6787±0.0136**

Table 2: The clustering accuracy rate(ACC)(%) on MNIST dataset for four views.

NMI are, the better the clustering performance of the corresponding method will be.

Experimental Environment We implement our method with the public toolbox of PyTorch and other partial data clustering methods with MATLAB on the same environment. All the experiments are run on the platform of Ubuntu Linux 16.04 with NVIDIA Titan Xp Graphics Processing Units (GPUs) and 64 GB memory size. Moreover, we use Adam (Kingma and Ba 2014) optimizer with the learning rate of 0.0001 and other default settings to train our model.

Experimental Results

ACC with Comparison Methods We conduct experiments with nine comparison algorithms on three public datasets and report experimental results in Table 1, 2, and 3, the best results are highlighted in **bold**. From these results, we can observe the following points: 1) The ACC results of our proposed method PVC-SSN are almost higher than all the other comparison methods. It is noteworthy that the ACC values of our method PVC-SSN is $2\% - 7\%$ higher than the highest competing algorithm iCmSC for different missing ratios on the MNIST dataset. The results demonstrate that our method can achieve better clustering than other comparison methods. 2) For partial data clustering, the methods GPVC and IMG are only suitable to the case of two-view data and our method can not only be easily extended to multi-view data, but also can achieve better clustering performance. The methods AMGL, RMSC and ConSC can achieve better clustering performance with complete multi-view data but are not satisfactory with the lack of partial multi-view data. 3) PVC-GAN achieves more remarkable clustering results than most of the other comparison methods. However, it focuses on generating the missing data via GAN and ignores the slim relationships among partial multi-view data, which causes the clustering performance low. Our method uses supervised contrastive loss to reduce the intra-cluster feature point distance while increasing the inter-cluster feature point distance, which makes the learned feature subspace more discriminative. 4) Most methods ignore the category information of partial multi-view data, which hampers further improvement of clustering performance. Whereas our method PVC-SSN could preserve category information embedded in multi-view data via feedback on clustering results and use it to guide the learning

Methods \ Missing rate	0.9	0.7	0.5	0.3	0.1
SC1	0.4633±0.0109	0.4943±0.0149	0.5272±0.0084	0.5402±0.0125	0.5871±0.0127
SC2	0.4775±0.0127	0.5113±0.0101	0.5446±0.0117	0.5871±0.0068	0.6266±0.0170
SC3	0.4863±0.0122	0.5188±0.0112	0.5664±0.0143	0.6114±0.0189	0.6613±0.0178
AMGL	0.6056±0.0489	0.6828±0.0564	0.7370±0.0281	0.7506±0.0320	0.7594±0.0211
RMSC	0.4642±0.0159	0.5293±0.0096	0.5925±0.0154	0.6507±0.0202	0.7154±0.0375
ConSC	0.5063±0.0325	0.5438±0.0272	0.5982±0.0246	0.6982±0.0481	0.7916±0.0299
GPVC	0.3238±0.0087	0.3077±0.0078	0.3419±0.0148	0.4236±0.0168	0.5370±0.0261
IMG	0.5350±0.0192	0.5455±0.0262	0.5457±0.0193	0.5529±0.0166	0.5633±0.0213
PVC-GAN	0.6546±0.0088	0.8517±0.0177	0.9069±0.0074	0.9342±0.0144	0.9425±0.0081
iCmSC	0.7610±0.0062	0.8205±0.0097	0.9158±0.0085	0.9450±0.0059	**0.9500±0.0064**
PVC-SSN	**0.8245±0.0057**	**0.8835±0.0132**	**0.9415±0.0046**	**0.9520±0.0066**	0.9485±0.0052

Table 3: The clustering accuracy rate(ACC)(%) on HW dataset for three views.

Missing Rate	0.9	0.7	0.5	0.3	0.1
PVC	0.5256	0.7016	0.8272	0.8912	0.9404
PVC-SSN	0.6032	0.8036	0.9156	0.9304	0.9616

Table 4: The ablation study of our method under different missing rates in terms of clustering accuracy rate on the BDGP dataset.

of subspace features, which is also why our method could outperform other competing methods.

Ablation Study In the subsection, we perform ablation studies on two versions of our method to study the importance of each component. We design two sub-modules *i.e.,* PVC and PVC-SSN to elaborate the role of the main components. PVC is the sub-module without the multi-view contrastive module and PVC-SSN is our method. We conduct the ablation experiments under the missing ratio $(0.1, 0.3, 0.5, 0.7, 0.9)$ on BDGP dataset, whose results are presented in Table 4. We can observe that the clustering performance is lowest without the multi-view contrastive module. When we embed the multi-view contrastive loss into the embedding space, the clustering performance is better because we maximize the agreement between feature points of the same cluster among partial multi-view data to learn a more consistent subspace representation. In fact, the multi-view contrastive encoder network contains category information and we feedback them to downstream training process, which can effectively guide the subspace feature learning. The experiment results demonstrate that the multi-view contrastive module plays an important role in our method.

Analysis of Subspace Feature In this subsection, we study the impact of different methods on the subspace features $\mathbf{Z}$. In order to show a t-SNE (Van der Maaten and Hinton 2008) visualization for feature embeddings in terms of different methods on the BDGP dataset with the missing ratio of 0.1, Figure 3 is given, in which different colors indicated different labels. The figure shows that our proposed PVC-SSN method has a more distinct and clear data distribution than other traditional clustering methods and most of deep clustering methods. This clearly confirms that the multi-view contrastive loss can make the subspace more friendly to the clustering task and can make the representation $\mathbf{Z}$ more discriminative.

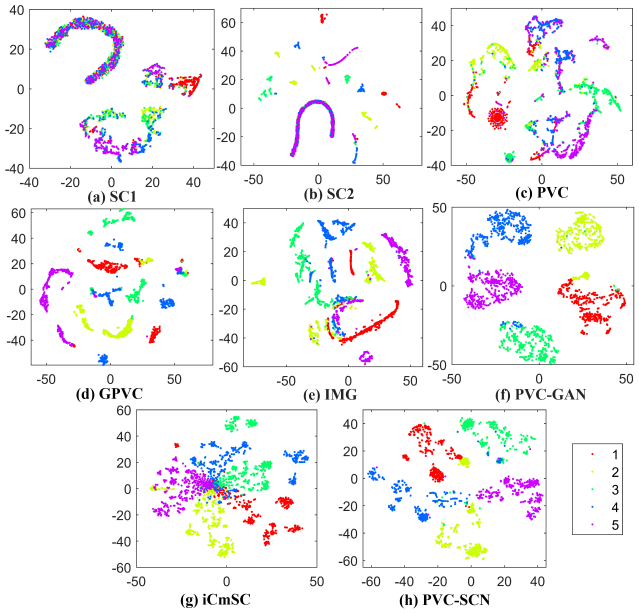

Figure 3: Visualization of the subspace features given by different methods with t-SNE on BDGP dataset with the missing ratio of 0.1, where (a) SC1, (b) SC2, (c) PVC, (d) GPVC, (e) IMG, (f) PVC-GAN, (g) iCmSC, and (h) PVC-SSN.

Impact of Different Parameters In our PVC-SSN model, λ_1 and λ_2 are two important hyper-parameters, which balances the role among different sub-modules. In order to test the parameter sensitivity, we conduct experiments on the influence of parameters on clustering performance under the missing ratio 0.9 on HW and MNIST datasets. In experiments, we set the ranges of parameters λ_1 and λ_2 to $[0.01, 0.1, 1, 10, 100]$. As shown in Fig. 4, we can observe

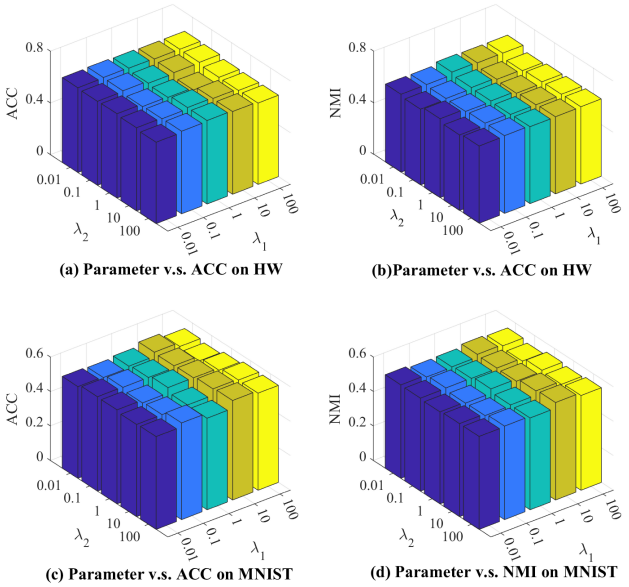

Figure 4: Parameters analysis on HW dataset and MNIST dataset with the missing ratio as 0.9. The clustering performance ACC and NMI on HW are shown in (a) and (b), and the clustering performance ACC and NMI on MNIST are shown in (c) and (d)

Methods	GPVC	IMG	PVC-GAN	iCmSC	PVC-SSN
Time(s)	10879	25721	74582	66010	46232

Table 5: The running time of all methods on BDGP dataset.

that the clustering performance of our proposed method changes a little with the fluctuation of parameters, which demonstrates our method PVC-SSN is robust for parameter changes. However, these parameters have a certain influence on clustering performance. As illustrated as Fig. 4, PVC-SSN achieves the best clustering result when $\lambda_1 = 0.01$ and $\lambda_2 = 0.1$ on HW dataset and it has the best clustering result when $\lambda_1 = 100$ and $\lambda_2 = 10$ on MNIST dataset.

Convergence Analysis In this subsection, we plot the change of overall loss value under the missing ratios 0.1 on BDGP, MNIST and HW datasets to investigate the convergence of our proposed method PVC-SSN. We conduct the convergence experiments with 2000 epochs on BDGP dataset and with 1000 epochs on MNIST dataset and HW dataset. In this case, the loss value on BDGP is recorded every 40 epochs and the loss value on MNIST and HW is recorded for per 20 epochs. As the curves depicted in Fig. 5, the clustering loss decreases and tends to stabilize as the epochs increase, which demonstrates that our proposed method has better convergence. Specifically, we can observe that on the BDGP dataset, the loss value drops rapidly in the first 1000 epochs, while on the MNIST dataset, the loss value drops rapidly in the first 600 epochs, and on the HW dataset, the loss value drops rapidly in the first 400 epochs. Therefore, our proposed optimization algorithm is reliable and converges quickly.

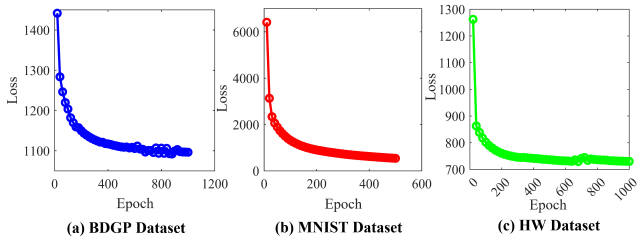

Figure 5: The convergence curves. Subfigure (a), (b), and (c) present the decrease of the overall loss for our proposed PVC-SSN method on BDGP, MNIST and HW dataset with the missing ratio of 0.1.

Time Complexity Comparison In this subsection, we report the running time in seconds of different deep partial multi-view clustering methods on BDGP dataset, as shown in Table 5. For a fair comparison, we test the running time of all deep clustering methods for 10000 epochs. From the results, we can see that the training time of our method is lower than both iCmSC and PVC-GAN. Therefore, the computation complexity of PVC-SSN is reasonable and its clustering performance is excellent.

Conclusion

In this paper, we propose a multi-view clustering network for partial multi-view data, named PVC-SSN, which embeds an effective contrastive learning module to explore the consistent feature from the same cluster among partial multi-view data and learn more discriminative subspace representation. Inspired by self-supervised learning, we use the clustering results obtained from the shared self-expression coefficient matrix to construct positive and negative sample pairs. The reported results demonstrate the superiority of our method when compared with other outstanding methods. This work focuses on partial multi-view clustering, and in the future, we will extend it to partial multi-view classification.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant 62176203 and Grant 62102306, the Fundamental Research Funds for the Central Universities, the Natural Science Basic Research Program of Shaanxi Province (Grant 2023-JC-YB-534), Shanghai Municipal Science and Technology Major Project (No.2018SHZDZX01), Key Laboratory of Computational Neuroscience and Brain-Inspired Intelligence (LCNBI) and ZJLab, and the Science and technology project of Xi'an (Grant 2022JH-JSYF-0009), the Natural Science Foundation of Shandong Province under Grant ZR202102180986, Guangxi Key Laboratory of Digital Infrastructure under Grant GXDIOP2023010, Natural Science Foundation of Guangdong Province, 2023A1515011845.

References

Andrew, G.; Arora, R.; Bilmes, J.; and Livescu, K. 2013. Deep canonical correlation analysis. In *ICML*, 1247–1255.

Cai, D.; He, X.; and Han, J. 2005. Document clustering using locality preserving indexing. *IEEE TKDE*, 17(12): 1624–1637.

Cai, X.; Wang, H.; Huang, H.; and Ding, C. 2012. Joint stage recognition and anatomical annotation of Drosophila gene expression patterns. *Bioinformatics*, 28(12): 16–24.

Chen, J.; Huang, A.; Gao, W.; Niu, Y.; and Zhao, T. 2023. Joint Shared-and-Specific Information for Deep Multi-View Clustering. *IEEE TCSVT*, 1–1.

Estévez, P. A.; Tesmer, M.; Perez, C. A.; and Zurada, J. M. 2009. Normalized mutual information feature selection. *IEEE TNN*, 20(2): 189–201.

Gao, H.; Nie, F.; Li, X.; and Huang, H. 2015. Multi-view subspace clustering. In *IEEE ICCV*, 4238–4246.

Gao, Q.; Lian, H.; Wang, Q.; and Sun, G. 2020. Cross-Modal Subspace Clustering via Deep Canonical Correlation Analysis. In *AAAI*, 3938–3945.

Jiang, G.; Peng, J.; Wang, H.; Mi, Z.; and Fu, X. 2022. Tensorial Multi-View Clustering via Low-Rank Constrained High-Order Graph Learning. *IEEE TCSVT*, 32(8): 5307–5318.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kumar, A.; Rai, P.; and Daume, H. 2011. Co-regularized multi-view spectral clustering. In *NeurIPS*, 1413–1421.

LeCun, Y. 1998. The MNIST database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*.

Li, S.-Y.; Jiang, Y.; and Zhou, Z.-H. 2014. Partial multi-view clustering. In *AAAI*, 1968–1974.

Li, Z.; Wang, Q.; Tao, Z.; Gao, Q.; and Yang, Z. 2019. Deep adversarial multi-view clustering network. In *IJCAI*, 2952–2958.

Liu, X.; Zhu, X.; Li, M.; Wang, L.; Tang, C.; Yin, J.; Shen, D.; Wang, H.; and Gao, W. 2018. Late fusion incomplete multi-view clustering. *IEEE TPAMI*, 41(10): 2410–2423.

Ng, A. Y.; Jordan, M. I.; and Weiss, Y. 2002. On spectral clustering: Analysis and an algorithm. In *NeurIPS*, 849–856.

Nie, F.; Li, J.; Li, X.; et al. 2016. Parameter-free auto-weighted multiple graph learning: A framework for multi-view clustering and semi-supervised classification. In *IJCAI*, 1881–1887.

Rai, N.; Negi, S.; Chaudhury, S.; and Deshmukh, O. 2016. Partial multi-view clustering using graph regularized NMF. In *ICPR*, 2192–2197. IEEE.

Shao, W.; He, L.; Lu, C.-t.; and Philip, S. Y. 2016. Online multi-view clustering with incomplete views. In *IEEE ICBD*, 1012–1017. IEEE.

Shao, W.; He, L.; and Philip, S. Y. 2015. Multiple Incomplete Views Clustering via Weighted Nonnegative Matrix Factorization with $\ell_{2,1}$ Regularization. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 318–334.

Sun, X.; Cheng, M.; Min, C.; and Jing, L. 2019. Self-Supervised Deep Multi-View Subspace Clustering. In *ACML*, 1001–1016.

Tao, Z.; Liu, H.; Li, S.; Ding, Z.; and Fu, Y. 2020. Marginalized Multiview Ensemble Clustering. *IEEE TNNLS*, 31(2): 600–611.

van Breukelen, M.; Duin, R. P.; Tax, D. M.; and Den Hartog, J. 1998. Handwritten digit recognition by combined classifiers. *Kybernetika*, 34(4): 381–386.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).

Wang, Q.; Cheng, J.; Gao, Q.; Zhao, G.; and Jiao, L. 2020a. Deep Multi-view Subspace Clustering with Unified and Discriminative Learning. *IEEE TMM*, 99(9): 1–11.

Wang, Q.; Ding, Z.; Tao, Z.; Gao, Q.; and Fu, Y. 2018a. Partial multi-view clustering via consistent GAN. In *IEEE ICDM*, 1290–1295.

Wang, Q.; Ding, Z.; Tao, Z.; Gao, Q.; and Fu, Y. 2021. Generative Partial Multi-View Clustering with Adaptive Fusion and Cycle Consistency. *IEEE TIP*, 99(9): 1–11.

Wang, Q.; Lian, H.; Sun, G.; Gao, Q.; and Jiao, L. 2020b. iCmSC: Incomplete Cross-Modal Subspace Clustering. *IEEE TIP*, 30: 305–317.

Wang, X.; Peng, D.; Yan, M.; and Hu, P. 2023a. Correspondence-free domain alignment for unsupervised cross-domain image retrieval. In *AAAI*, 10200–10208.

Wang, Y.; Chang, D.; Fu, Z.; Wen, J.; and Zhao, Y. 2023b. Incomplete Multiview Clustering via Cross-View Relation Transfer. *IEEE TCSVT*, 33(1): 367–378.

Wang, Y.; Wu, L.; Lin, X.; and Gao, J. 2018b. Multiview spectral clustering via structured low-rank matrix factorization. *IEEE TNNLS*, 29(10): 4833–4843.

Wen, J.; Xu, Y.; and Liu, H. 2018. Incomplete multiview spectral clustering with adaptive graph learning. *IEEE TC*.

Wen, J.; Zhang, Z.; Fei, L.; Zhang, B.; Xu, Y.; Zhang, Z.; and Li, J. 2022. A survey on incomplete multiview clustering. *IEEE TCybe*, 53(2): 1136–1149.

Xia, R.; Pan, Y.; Du, L.; and Yin, J. 2014. Robust multi-view spectral clustering via low-rank and sparse decomposition. In *AAAI*, 2149–2155.

Xu, C.; Tao, D.; and Xu, C. 2015. Multi-view learning with incomplete views. *IEEE TIP*, 24(12): 5812–5825.

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

Zhang, C.; Cui, Y.; Han, Z.; Zhou, J. T.; Fu, H.; and Hu, Q. 2020. Deep partial multi-view learning. *IEEE TPAMI*.

Zhang, C.; Hu, Q.; Fu, H.; Zhu, P.; and Cao, X. 2017. Latent multi-view subspace clustering. In *IEEE CVPR*, 4279–4287.

Zhao, H.; Liu, H.; and Fu, Y. 2016. Incomplete multi-modal visual data grouping. In *IJCAI*, 2392–2398.