mathematics

MDPI

Article

Cluster Complementarity and Consistency Mining for Multi-View Representation Learning

Yanyan Wen [1,2] and Haifeng Li [3,*]

[1] School of Management, Xi'an Jiaotong University, Xi'an 710049, China; wenyanyan@xjtu.edu.cn
[2] Business School, Qinghai Institute of Technology, Xining 810016, China
[3] School of Mathematics and Statistics, Fuyang Normal University, Fuyang 236037, China
* Correspondence: lihaifeng@fynu.edu.cn

Abstract

With the advent of the big data era, multi-view clustering (MVC) methods have attracted considerable acclaim due to their capability in handling the multifaceted nature of data, which achieves impressive results across various fields. However, two significant challenges persist in MVC methods: (1) They resort to learning view-invariant information of samples to bridge the heterogeneity gap between views, which may result in the loss of view-specific information that contributes to pattern mining. (2) They utilize fusion strategies that are susceptible to the discriminability of views, i.e., the concatenation and the weighing fusion of cross-view representations, to aggregate complementary and consistent information, which is difficult to guarantee semantic robustness of fusion representations. To this end, a simple yet effective cluster complementarity and consistency learning framework (CommonMVC) is proposed for mining patterns of multiview data. Specifically, a cluster complementarity learning is devised to endow fusion representations with discriminate information via nonlinearly aggregating view-specific information. Meanwhile, a cluster consistency learning is introduced via modeling instance-level and cluster-level partition invariance to coordinate the clustering partition of various views, which ensures the robustness of multi-view data pattern mining. Seamless collaboration between two components effectively enhances multi-view clustering performance. Finally, comprehensive experiments on four real-world datasets demonstrate CommonMVC establishes a new state-of-the-art baseline for the MVC task.

Keywords: multi-view clustering; complementarity and consistency; nonlinear fusion

MSC: 68T10

check for updates

Academic Editors: Shuo Yu and Feng Xia

Received: 1 July 2025
Revised: 29 July 2025
Accepted: 4 August 2025
Published: 5 August 2025

Citation: Wen, Y.; Li, H. Cluster Complementarity and Consistency Mining for Multi-View Representation Learning. *Mathematics* **2025**, *13*, 2521. https://doi.org/10.3390/math13152521

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

With the proliferation of multimedia technologies, a vast amount of data generated by various sensors and extractors has emerged in recent years [1,2]. For instance, in the traffic analysis task, data usually are collected from ultrasonic radars, optical cameras, and infrared radars. In healthcare, data can be obtained from various diagnostic tools such as MRI scans, CT scans, and X-rays. These data are commonly called multi-view data, and each view exhibits consistency that expresses common properties of samples and complementarity that expresses specific properties of samples. Effective aggregation of consistent and complementary information can endow data with discriminative and expressive abilities, which greatly facilitate the mining of data patterns in the downstream tasks [3,4].

As one of the most important multi-view information fusion paradigms, multi-view clustering (MVC) has received increasing attention, which is divided into two categories based on differences among fusion strategies of consistent and complementary information, i.e., the multi-view concatenation fusion method [5–8] and the multi-view weighing fusion method [9–12]. The former directly concatenates features from different views into a single fusion representation to fuse consistent and complementary information for clustering. The latter assigns different weights to features of each view based on their importance and reliability in fusing consistent and complementary information, ensuring that the most informative views contribute more to the final clustering result.

Although existing MVC methods have made encouraging progress in mining patterns of multi-view data, there are still two issues. Firstly, most MVC methods [13–16] impose alignment constraints, such as contrastive alignment and adversarial alignment, on view-specific representations to bridge the heterogeneity gap between views for aggregating consistent and complementary information. For example, MGCC proposes a multi-view granular ball contrastive clustering method via maximizing similarities between data within the same granular ball and minimizing similarities between data in different granular balls, which enforces the local structure invariance for clustering [17]. However, the essence of these alignment constraints is to explore inter-view invariant information, i.e., consistency, which may suppress complementarity information mining of multi-view data and cause the loss of useful information. Secondly, most MVC methods [12,18–20] utilize fusion strategies, i.e., the direct concatenation or the weighed combination of view-specific representations, to obtain view-common representations of multi-view data. For example, DMAC-SI concatenates view-specific representations with semantic invariance in the mirror encoder-decoder framework and then imposes a data-driven partitioning strategy on fusion representations for obtaining clustering results [21]. Such a manner, which is susceptible to representation discriminability in each view, may not guarantee semantic robustness of view-common representations.

In the paper, a simple and effective cluster complementarity and consistency leaning (CommonMVC) is proposed to solve the above problems. Specifically, CommonMVC defines a view-specific representation learning and cluster partitioning component via deploying encoding-decoding networks with clustering layers for each view to obtain vanilla latent representations and soft assignments of data. Then, CommonMVC designs a cluster complementarity learning via the nonlinear fusion functions to learn view-common representations, which can fully complement information between views and avoid the nuisance of certain views with fuzzy clustering structures. These nonlinear fusion functions are derived from maximizing mutual information between views, which are implemented using the multi-layer perceptron networks. Meanwhile, CommonMVC designs a cluster consistency learning via modeling instance-level and cluster-level partition invariance. The former ensures that the same sample from different views belongs to the same cluster, while the latter ensures that the same cluster in different views has a similar sample distribution. This dual-level invariance enhances the consistency and robustness of multi-view data pattern mining. In CommonMVC, the seamless collaboration of the three components in CommonMVC yields improved multi-view clustering performance with more accurate, comprehensive, and consistent clustering results in multi-view data. Ultimately, experiment results on four real-world datasets demonstrate that CommonMVC achieves state-of-the-art performance in MVC.

The main contributions of CommonMVC are threefold:

- A complementarity-leaning strategy is proposed to nonlinearly aggregate view-specific information, which ensures robustness of fusion representations.

- A consistency leaning strategy is devised via modeling instance-level and cluster-level partition invariance, to alleviate nuisances of some views with fuzzy cluster structures, which effectively enhances intrinsic pattern mining.
- Extensive evaluations conducted on four benchmarks verify that CommonMVC sets a new standard for MVC tasks compared with baseline methods.

The paper proceeds as follows: Section 2 constructs related work of CommonMVC. Section 3 constructs the methodological framework, detailing our novel architecture and adaptive mechanisms. Section 4 conducts rigorous multidimensional evaluation across benchmarks, comparative analysis, and ablation studies. Section 5 synthesizes core contributions, discusses practical implications, and delineates emergent research trajectories.

2. Related Work

Multi-view clustering refers to the methodology in data analysis and machine learning that utilizes information from multiple sources or feature sets to enhance clustering accuracy and robustness. Based on the paradigm of information fusion, MVC methods are divided into multi-view concatenation fusion methods and multi-view weighing fusion methods.

2.1. Multi-View Concatenation Fusion Method

Multi-view concatenation fusion methods combine features from different views into a single comprehensive representation by directly concatenating them, enabling unified data processing and clustering. For instance, 2CMV concatenates consistent and complementary information extracted via the matrix factorization to learn fusion representations and then performs vanilla k-means on fusion representations to obtain clustering results of multi-view data [22]. MFK concatenates consistent and complementary information within the mutual guidance between view-common cluster partitioning and view-specific coefficient matrices learning, and then employs k-means on fusion representations to obtain clustering results of multi-view data [23]. MvAGC concatenates smooth view-specific representations learned via self-expression layers with the graph filter and then utilizes spectral clustering on the consensus matrix of fusion representations for capturing clustering structures [24]. CSMSC concatenates consistent and complementary information between views disentangled via view-common and view-specific self-expression matrices for performing spectral clustering to explore data structures [25]. DMCAG defines an anchor multi-view subspace clustering to learn the consensus matrix between views, and then utilizes spectral self-supervised learning to align view-specific clustering assignment for mining patterns of data [26]. DMIM designs an over-clustering encoding-decoding network via maximizing the dependence between representations in different views and minimizing the dependence between representations and data in the same views and then performs k-means on weighted fusion representations to explore clustering patterns [11]. SDMVC concatenates view-specific representations learned via a self-supervised clustering consistency strategy to fuse complementary and consistent information, along with an end-to-end pattern mining [12].

2.2. Multi-View Weighing Fusion Method

Multi-view weighing fusion methods assign different weights to each view based on their importance and reliability, combining them to capture both consistent and complementary information for more accurate and robust clustering. For instance, CoMVC conducts a contrastive encoding-decoding network via weighted aligning view-specific complementary information to learn view-common representations and then utilizes the Cauchy-Schwarz divergence to explore structure divergences between data [27]. MVaDE conducts a variational generation network from hidden categories to real data for weighted

aggregation of consistent and complementary information, where the mixture of Gaussian distributions is used to infer cluster partitions of multi-view data [7]. MSEC obtains an inter-view consensus matrix for clustering via weighted fusion of view-specific consensus matrices that are learned via view-specific metric functions, where inter-view and intra-view structure invariant constraints are used to enhance the aggregation of consistent and complementary information [8]. CMSC fuses information of views based on view importance to learn the consensus matrix in a shared subspace where the correlations between views are maximized via the canonical correlation analysis and then performs spectral clustering to explore data structures [28]. COMIC aligns view-specific graphs to aggregate consistent and complementary information and then conducts weighted partitions in a view-specific graph to mine patterns of multi-view data [10]. CMIB designs an information bottleneck encoding-decoding network to learn robust view-common representations via weighting complementary, consistent, and superfluous information among views and then performs k-means on view-common representations to obtain final clustering results [9]. AE2-Nets designs an encoding-decoding degradation network to fuse consistent and complementary information for learning fusion representations and then performs k-means to obtain final clustering results [29]. DealMVC designs a dual contrastive multi-view clustering network via aligning inter-view and intra-view feature-level similarity graphs and class-level pseudo-label graph and then enforces intra-view pseudo-label graphs to be as similar as possible to obtain clustering results [13]. VCGA devises an information-decoupled multi-view graph clustering method via exploring view-specific graphs and the consensus graphs with the help of the local and global structure exploration, which fully explores the local manifold for mining patterns of multiview data [18]. HCLS CGL designs a confidence neighbor multi-view weighing fusion via weighting the group-wise structures between data to boost the aggregation of consistent and complementary structures across different views by leveraging the confidence of neighboring relationships [30]. STMVC proposes an uncertainty refinement strategy to trustily separate multi-view representations significantly belonging to different clusters to fuse well-separated view-specific representations into high-quality discriminative representations for clustering, where the discriminative representations were weighted on the basis of view-specific uncertainties according to the Dempster-Safer theory of evidence [14]. DMAC proposes an anchor graph alignment strategy to match multi-view representations of the view-average discriminative representations for clustering by introducing a perturbation-injected learnable anchor graph, where the anchor graph convolution clustering distributions instructed the consistency refinement of discriminative multi-view representations for efficient cluster structure mining [15].

In multi-view clustering, existing fusion strategies mainly fall into concatenation fusion and weighted fusion. Concatenation fusion directly connects features from different views, which raises the dimensionality and brings in redundant info. Weighted fusion assigns weights to views based on their importance and then combines them linearly, but it has a subjective weight-allocation process and is sensitive to view-quality differences. Nonlinear fusion, as in CommonMVC, surpasses both. Using nonlinear transformation functions, CommonMVC can uncover complex relationships between views. This makes the model more robust and better at fitting data structures, improving clustering accuracy. So, nonlinear fusion is a more effective, flexible, and robust solution than concatenation and weighted fusion.

3. The Proposed Method

Consider the task of clustering a multi-view data set of n samples with V views $\{x_i\}_{i=1}^n = \{x_i^1 \in \mathbb{R}^{D^1}, \ldots, x_i^V \in \mathbb{R}^{D^V}\}_{i=1}^n$, where x_i^v denotes the v-th view data of the i-th

sample x_i and D^v denotes the feature dimension in the v-th view. A cluster complementarity and consistency leaning (CommonMVC) is invented to achieve MVC, which consists of view-specific representation learning and cluster partitioning, cluster complementarity learning, and cluster consistency learning, as shown in Figure 1.

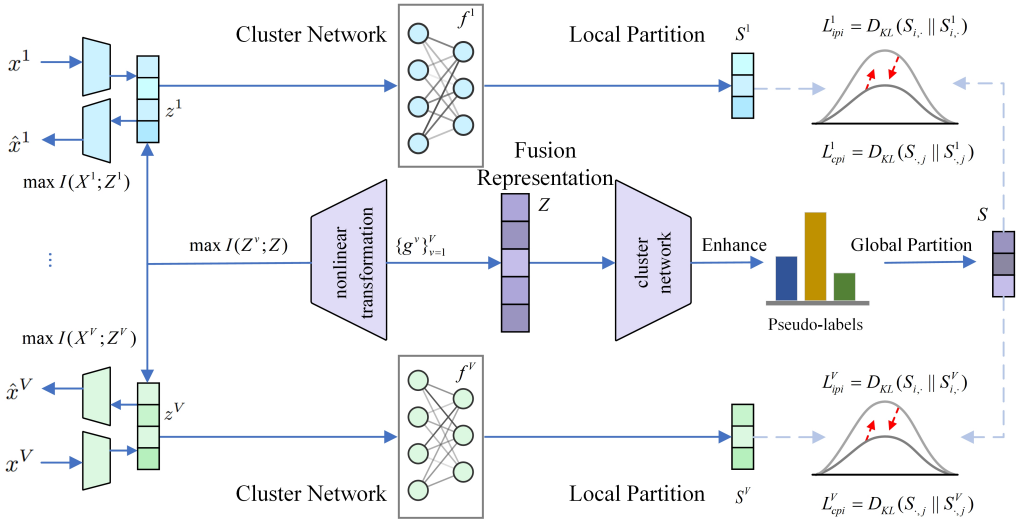


Figure 1. The illustration of CommonMVC. Given multi-view data $= \{x_i^1, \ldots, x_i^V\}_{i=1}^n$, CommonMVC generates view-specific representations $= \{z_i^1, \ldots, z_i^V\}_{i=1}^n$ via deep autoencoders, i.e., $z_i^v = Enc^v(x_i^v; \theta^v)$ and $\bar{x}_i^v = Dec^v(z_i^v; \phi^v)$, and then obtains view-specific cluster partitions $\{s_i^1, \ldots, s_i^V\}_{i=1}^n$ from view-specific representations via clustering network $\{f^v\}_{v=1}^V$. Meanwhile, CommonMVC obtains fusion representation $\{Z_i\}_{i=1}^n$ via nonlinear transformation functions $\{g^v\}_{v=1}^V$ and generates the global partition S via the clustering network MLP$(\cdot)$. Finally, CommonMVC maximizes the mutual information $I(X^v; Z^v)$ to optimize the view-specific representation learning and cluster partitioning, maximizes the mutual information $I(Z; Z^v)$ to achieve the cluster complementarity learning, and performs instance-level and cluster-level partition invariance learning via minimizing $D_{KL}(S_{i,\cdot}, S_{i,\cdot}^v)$ and $D_{KL}(S_{\cdot,j}, S_{\cdot,j}^v)$, respectively.

3.1. View-Specific Representation Learning and Cluster Partitioning

Generally, each view, describing different aspects of the same samples, contains view-specific complementary information, which can greatly enhance the clustering performance. Thus, view-specific representation learning and cluster partitioning are designed to fully explore and capture intrinsic information hidden in each view.

Representation Learning. Inspired by effectiveness in previous works [31–33], deep autoencoders are utilized to learn latent representations of heterogeneous multi-view data with high dimensions. The deep autoencoder for each view works as follows: the encoder compresses the original data into a lower-dimensional latent space, capturing the essential features while discarding noise and redundancy. The decoder then attempts to reconstruct the original data from this latent representation. This process helps the model learn robust and meaningful features.

Specifically, given the v-th view data of the i-th sample x_i^v, the encoder Enc^v is used to transform data x_i^v into the latent representation z_i^v, i.e., $z_i^v = Enc^v(x_i^v; \theta^v)$, where θ^v denotes the encoder parameter in the v-th view. Meanwhile, the decoder Dec^v is used to generate reconstruction data $\bar{x}_i^v$ via z_i^v, i.e., $\bar{x}_i^v = Dec^v(z_i^v; \phi^v)$, where ϕ^v denotes the decoder parameter in the v-th view. To ensure that the learned representations retain as much useful information as possible from the original data, we maximize the mutual

information between the original data and the corresponding representations in each view, as follows:

$$\begin{aligned} L_{base} &= -\sum_{v=1}^{V} \max I(X^v; Z^v) \\ &= -\sum_{v=1}^{V} \int dx^v dz^v p(x^v, z^v) \log \frac{p(x^v, z^v)}{p(x^v)p(z^v)} \\ &\leq \sum_{v=1}^{V} \int dx^v p(x^v) \int dz p(z^v|x^v) \log q(x^v|z^v) \end{aligned} \tag{1}$$

where $I(X^v; Z^v)$ represents the mutual information between the original data X^v and its latent representation Z^v in the v-th view. Mutual information quantifies the amount of information obtained about one random variable through the other. By maximizing it, we ensure that the latent representations Z^v retain as much information as possible from the original data X^v, which is crucial for preserving the essential characteristics and patterns in the data.

Cluster Partitioning. To avoid suboptimal representation for clustering, the parameterized nonlinear mappings based on Student's t-distribution are constructed on the top of encoders in each view, which generates soft cluster assignments of samples in an end-to-end manner, i.e., $f^v(Z^v; C^v) : Z^v \rightarrow S^v$, where $C^v = \{c_1^v, c_2^v, \ldots, c_k^v\}$ denotes the learnable parameters and S^v denotes soft cluster assignments of samples in the v-th view. The generation process of S^v is as follows:

$$s_{ij}^v = \frac{(1 + \left\| z_i^v - c_j^v \right\|^2)^{-1}}{\sum_{l=1}^{k} (1 + \left\| z_i^v - c_l^v \right\|^2)^{-1}} \tag{2}$$

where s_{ij}^v is the j-th element of S_i^v and represents the probability of assigning the i-th sample to the j-th cluster in the v-th view. Cluster partitioning utilizes Student's t-distribution to produce smooth and robust cluster assignments, where samples closer to a cluster center have higher assignment probabilities.

3.2. Cluster Complementarity Learning

In general, some information aggradation policies, i.e., concatenated and weighed combinations, are used to generate fusion representations between views via aggregating view-specific representations. However, the concatenated combination, i.e., $Z = [Z^1, Z^2, \ldots, Z^V]$, has two significant drawbacks that may distort the final fused representation and ultimately degrade performance in downstream tasks. First, it introduces view redundancy, leading to the inclusion of repetitive or irrelevant information. Second, it is susceptible to the influence of noisy views with fuzzy data structures. The weighted combination, i.e., $Z = W^1 Z^1 + W^2 Z^2 +, \ldots, +W^V Z^V$, where W denotes the weight matrix, is commonly regarded as a linear mapping from view-specific representations to fusion representations. Such a linear mapping between variables cannot capture complementary information, i.e., the inseparable clusters in one variable are still inseparable in other variables. Conversely, if there is complementarity between variables, the complementary information can be delivered by a nonlinear mapping.

Rooted in the above discussions, the simple and effective nonlinear transformation functions are designed to learn fusion representations, i.e., $Z^v = f^v(Z)$, where f^v denotes the nonlinear mapping function of the v-th view. Then, the mutual information between Z and Z^v is maximized to optimize parameters of g^v, which effectively aggregates inter-view complementarity learning of clustering structures, as follows:

$$\begin{aligned} L_{com} &= -\sum_{v=1}^{V} \max I(Z;Z^v) = -\sum_{v=1}^{V} \int dz^v dz p(z^v,z) \log \frac{p(z^v,z)}{P(z^v)P(z)} \\ &= -\sum_{v=1}^{V} \int dz^v dz p(z^v,z) \log \frac{p(z^v|z)}{p(z^v)} \\ &= -\sum_{v=1}^{V} H(z^v) + \int dz p(z) \int dz^v p(z^v|z) \log p(z^v|z) \\ &\leq \int dz^v p(z^v) \int dz p(z|z^v) \log q(z^v|z) \end{aligned} \tag{3}$$

where $q(z^v|z)$ is the inferring distribution from z to z^v, which can be implemented by the nonlinear mapping function g^v.

Cluster complementarity learning utilizes the nonlinear transformation functions to capture fusion representations with more discriminative clustering structure than that of view-specific representations and can effectively relieve interference of views with fuzzy clustering structures in fusing complementary patterns, ensuring the cluster robustness in learning multi-view fusion representations.

3.3. Cluster Consistency Learning

Due to the absence of label information in multi-view clustering, it is not possible to determine which view's clustering partition is precise. However, multi-view clustering consistency can help mitigate these challenges because it ensures that different views provide similar clustering predictions, which reduces the likelihood of relying on any single, potentially inaccurate view. By aligning the clustering results across multiple views, the overall accuracy and robustness of the clustering outcomes are enhanced. To ensure this consistency, we propose a novel cluster consistency learning via utilizing a global clustering partition to effectively harmonize the clustering predictions across different views. Specifically, multi-view cluster consistency learning consists of instance-level partition invariance and cluster-level partition invariance. The former refers to the fact that the same sample from different views should belong to the same label, and the latter refers to the fact that the same cluster in different views should satisfy the same sample distribution.

To achieve two-partition invariance in the cluster consistency learning, a global semantic distribution is obtained as follows. A parameterized MLP is constructed to generate global semantics of fusion representations, i.e., $MLP(Z;C) : Z \rightarrow S$, where $C = \{c_1, c_2, \ldots, c_k\}$ stands for cluster prototypes of fusion representations and S denotes the global clustering partition:

$$S = MLP(Z;C) = \delta(\varepsilon(Z,C)) \tag{4}$$

where $\varepsilon(\cdot)$ stands for the membership relationship between data and clusters:

$$q_{ij} = \varepsilon(z_i, c_j) = \frac{1}{1 + \left\| z_i - c_j \right\|_2^2} \tag{5}$$

where q_{ij} stands for the similarity degree of the i-th sample and the j-th cluster. $\delta(\cdot)$ transforms the similarities to the clustering partition:

$$s_{ij} = \delta(s_i) = \frac{\left(q_{ij} / \sum_j q_{ij}\right)^2}{\sum_j \left(q_{ij} / \sum_j q_{ij}\right)^2} \tag{6}$$

The Kullback–Leibler divergence between the global clustering partition of fusion representations $S_{i,\cdot}$ and local clustering partitions of view-specific representations $S^v_{i,\cdot}$ is minimized to perform instance-level partition invariance learning, where $S_{i,\cdot}$ and $S^v_{i,\cdot}$ denote the i-th row of the global clustering partition and local clustering partitions, respectively.

$$L_{ipi} = -\sum_{v=1}^{V}\sum_{i=1}^{n}\sum_{j=1}^{k} s_{ij}\log\frac{s_{ij}}{s^v_{ij}} \tag{7}$$

Meanwhile, the Kullback–Leibler divergence between the global sample distribution of clusters $S_{\cdot,i}$ and local sample distribution of clusters $S^v_{\cdot,i}$ is minimized to perform cluster-level partition invariance learning, where $S_{\cdot,i}$ and $S_{\cdot,i}$ denote the i-th column of the global clustering partition and local clustering partitions, respectively.

$$L_{cpi} = -\sum_{v=1}^{V}\sum_{j=1}^{k}\sum_{i=1}^{n} s_{ij}\log\frac{s_{ij}}{s^v_{ij}} \tag{8}$$

After training the model, final clustering results of samples are obtained as follows:

$$y_i = \arg\max_j(s_{ij}) \tag{9}$$

3.4. The Loss Function

CommonMVC conducts the loss function L to optimize the cluster complementarity learning and cluster consistency learning, as follows:

$$L = L_{base} + \alpha L_{com} + \beta L_{ipi} + \gamma L_{cpi} \tag{10}$$

where α, γ, and β are parameters. L_{com} is the complementary loss that guides cluster complementarity learning. L_{ipi} and L_{cpi} are the consistent losses that guide cluster consistency learning. That is, L trains CommonMVC to fully explore complementarity and consistency hidden in multi-view data, which effectively enhances intrinsic pattern mining.

The model analysis. The analysis of why L_{ipi} and L_{cpi} can guarantee inter-view-consistent clustering partitions is as follows: For the v-th and the l-th views, given the optimization errors ϵ_v and ϵ_l of L^v_{ipi} and L^j_{ipi}, respectively, we have:

$$0 \le KL(S_{i,\cdot} \parallel S^v_{i,\cdot}) = \sum_{i=1}^{n}\sum_{j=1}^{K} s_{ij}\log\frac{s_{ij}}{s^v_{ij}} \le \varepsilon_v \tag{11}$$

$$0 \le KL(S_{i,\cdot} \parallel S^l_{i,\cdot}) = \sum_{i=1}^{n}\sum_{j=1}^{K} s_{ij}\log\frac{s_{ij}}{s^l_{ij}} \le \varepsilon_l \tag{12}$$

via subtracting Equation (11) from Equation (12), having:

$$-\varepsilon_v \le \sum_{i=1}^{n}\sum_{j=1}^{K} s_{ij}\log\frac{s^v_{ij}}{s^l_{ij}} \le \varepsilon_l \tag{13}$$

When s_{ij} is fixed, and ϵ_v and ϵ_l decrease in the optimization precessing:

$$\sum_{i=1}^{n}\sum_{j=1}^{K} s_{ij}\log\frac{s^v_{ij}}{s^l_{ij}} \to 0 \tag{14}$$

Thus, $S^v_{i,\cdot}$ and $S^l_{i,\cdot}$ tend to be consistent.

Consider ξ_v and ξ_l as the optimization error of L^v_{cpi} and L^j_{cpi}, respectively, having:

$$0 \leq KL(S_{\cdot,j}||S^v_{\cdot,j}) = \sum_{j=1}^{K}\sum_{i=1}^{n} s_{ij}\log\frac{s_{ij}}{s^v_{ij}} \leq \xi_v \tag{15}$$

$$0 \leq KL(S_{\cdot,j}||S^l_{\cdot,j}) = \sum_{j=1}^{K}\sum_{i=1}^{n} s_{ij}\log\frac{s_{ij}}{s^l_{ij}} \leq \xi_l \tag{16}$$

via subtracting Equation (15) from Equation (16), having:

$$-\xi_v \leq \sum_{j=1}^{K}\sum_{i=1}^{n} s_{ij}\log\frac{s^v_{ij}}{s^l_{ij}} \leq \xi_l \tag{17}$$

When s_{ij} is fixed, and ξ_v and ξ_l gradually decrease in the optimization precessing:

$$\sum_{j=1}^{K}\sum_{i=1}^{n} s_{ij}\log\frac{s^v_{ij}}{s^l_{ij}} \to 0 \tag{18}$$

Thus, $S^v_{\cdot,j}$ and $S^l_{\cdot,j}$ tend to be consistent. Cluster consistent learning ensures multi-view clustering consistency via utilizing S to align rows and columns of each view S^v.

4. Experimental Evaluation

4.1. Setup

Datasets. Four public multi-view datasets, i.e., Handwritten, ORL, LandUse-21, and Scene-15, are utilized to evaluate the performance of CommonMVC. Handwritten contains 2000 samples from 10 classes, where pixel averages in 2 × 3 windows and profile correlations are used as two views. ORL comprises 400 samples belonging to 40 classes where intensity of gray level and Gabor descriptors are used as two views. LandUse-21 includes 2100 samples categorized into 21 classes, where PHOG and LBP features are used as two views. Scene-15 consists of 4485 samples spanning 15 classes, where PHOG and GIST features are used as two views. The detailed information is shown in Table 1.

Table 1. The information of datasets.

Dataset	Samples	View	Class	Content
Handwritten	2000	2	10	Handwritten Digits
ORL	400	2	40	Human Faces
LandUse-21	2100	2	21	Satellite Images
Scene-15	4485	2	15	Scene Images

Evaluation Metrics: To evaluate the clustering performance, three common metrics, i.e., ACC, NMI, and ARI, are used in the experiments [31]. ACC represents the rate at which samples are correctly partitioned. NMI represents similarity between the prediction distribution and the ground-truth distribution of data. ARI represents the purity of predictions. The larger values of the ACC, NMI, and ARI mean the superior clustering performance.

Comparison methods. In the experiment, fourteen comparison methods are used, including MVaDE [9], CoMVC [10], DealMVC [7], GCFAggMVC [11], SDMVC [12], DCCA [34], DCCAE [35], AE^2-Nets [8], MSEC [28], MGCC [17], DMAC [19], and STMVC [14]. To ensure fair comparison, all experiments are conducted under identical hardware and software environments, with hyperparameter configurations strictly following the specifications reported in each method's original publication.

Implementation details. CommonMVC, implemented using PyTorch 2.0.1, was employed in the experiments, where the vector features of the samples were first normalized to the $[0, 1]$ range before being input into the model for training. The training process utilized the Adam optimizer with a batch size of 100, maintaining a learning rate of 10^{-3} across all CommonMVC modules. The trade-off parameters α, β, and γ were consistently set to $[0.1, 1]$, respectively, for all datasets. To ensure optimal performance of the comparison methods, a grid search strategy was applied to their trade-off parameters. Each experiment is conducted five times, with the average results reported as the final outcomes to ensure a fair comparison.

4.2. Clustering Performance Evaluation

Tables 2 and 3 illustrate the quantitative clustering results of CommonMVC and fourteen comparison methods on four datasets in terms of ACC, NMI, and ARI. The results show that CommonMVC contains the optimal clustering performance and sets a new baseline in the multi-view clustering task. The seasons have two aspects: (1) CommonMVC designs a cluster complementarity learning via the nonlinear fusion functions to learn view-common representations, which fully captures complementary information between views and avoids the nuisance of certain views with fuzzy clustering structures. (2) CommonMVC designs a cluster consistency learning via defining a relaxed alignment on the semantic level rather than the representation level, which alleviates the tendency of view-specific representations to be the same while capturing consistent information between views during the optimization process. In addition, there are also two observations. First, the clustering results of MVC methods are better than those obtained by FeatConcate, which proves the necessity of developing multi-perspective clustering methods. Second, the performance of one-stage MVC methods outperforms two-stage MVC methods in pattern mining of data. The reason is that one-stage MVC methods capture clustering-friendly information of multi-view data, rather than common information captured by two-stage MVC methods.

Table 2. The average clustering results on Handwritten and ORL datasets.

Dataset	**Handwritten**			**ORL**		
Metric	***ACC***	***NMI***	***ARI***	***ACC***	***NMI***	***ARI***
FeatConcate	0.610	0.607	0.553	0.571	0.753	0.477
DCCA	0.663	0.660	0.614	0.597	0.778	0.502
DCCAE	0.692	0.670	0.633	0.594	0.775	0.499
AE^2-Nets	0.815	0.714	0.667	0.688	0.757	0.514
MVaDE	0.888	0.808	0.776	0.695	0.736	0.504
SiMVC	0.830	0.761	0.698	0.692	0.756	0.526
SDMVC	0.899	0.821	0.801	0.610	0.756	0.504
GCFAggMVC	0.828	0.717	0.666	0.650	0.835	0.509
MSEC	0.862	0.827	0.698	0.647	0.770	0.551
DealMVC	0.813	0.718	0.642	0.137	0.329	0.044
SGDMC	0.904	0.837	0.818	0.702	0.825	0.551
MGCC	0.893	0.820	0.806	0.688	0.819	0.552
DMAC	0.888	0.817	0.798	0.686	0.804	0.527
STMVC	0.910	0.840	0.820	0.692	0.818	0.546
CommonMVC	**0.912**	**0.848**	**0.823**	**0.711**	**0.848**	**0.580**

Figure 2 presents the outcomes of the Nemenyi test. A detailed perusal of the figure reveals that the average rank of the CommonMVC method surpasses those of the other methods across all four datasets. This notable ranking advantage carries substantial statistical weight. It strongly suggests that the CommonMVC method holds a significant edge

over its counterparts. The consistent excellence of the CommonMVC method across various datasets makes it stand out distinctively among numerous approaches. Consequently, it offers a more precious option for relevant studies and applications.

Table 3. The average clustering results on LandUse and Scene datasets.

Dataset	**LandUse-21**			**Scene-15**		
Metric	***ACC***	***NMI***	***ARI***	***ACC***	***NMI***	***ARI***
FeatConcate	0.123	0.161	0.036	0.208	0.304	0.116
DCCA	0.155	0.232	0.044	0.362	0.289	0.109
DCCAE	0.156	0.244	0.044	0.364	0.298	0.115
AE^2-Nets	0.248	0.304	0.104	0.261	0.304	0.121
MVaDE	0.225	0.225	0.094	0.378	0.299	0.118
SiMVC	0.245	0.258	0.096	0.377	0.294	0.126
SDMVC	0.238	0.229	0.120	0.386	0.213	0.126
GCFAggMVC	0.240	0.242	0.115	0.286	0.205	0.124
MSEC	0.234	0.253	0.098	0.285	0.233	0.154
DealMVC	0.180	0.192	0.065	0.278	0.226	0.140
SGDMC	0.243	0.266	0.110	0.293	0.281	0.151
MGCC	0.204	0.220	0.098	0.253	0.274	0.128
DMAC	0.238	0.255	0.116	0.300	0.301	0.142
STMVC	0.223	0.242	0.103	0.292	0.307	0.146
CommonMVC	**0.259**	**0.280**	**0.120**	**0.310**	**0.312**	**0.152**

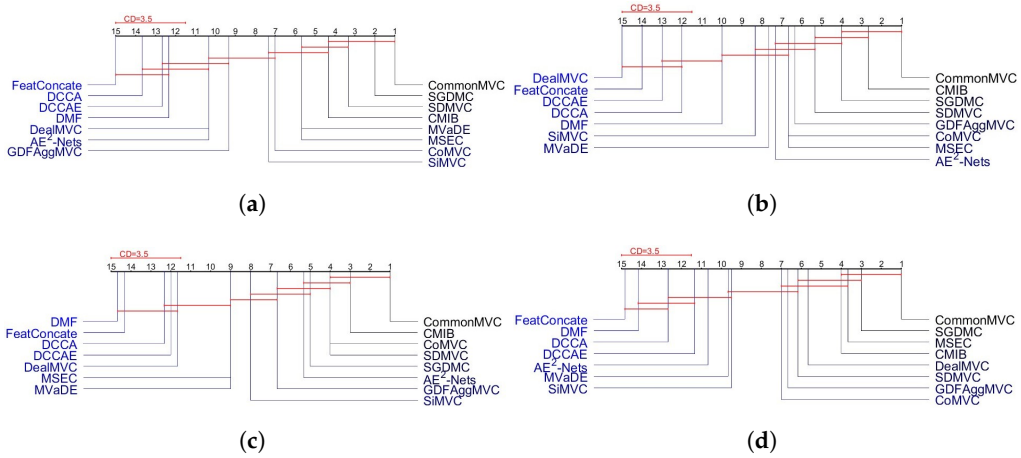


Figure 2. The Nemenyi test on four datasets. (**a**) Nemenyi test on Handwritten; (**b**) Nemenyi test on ORL; (**c**) Nemenyi test on LandUse-21; (**d**) Nemenyi test on Scene-15.

4.3. Ablation Analysis

This section conducts loss ablation and architecture ablation on Handwritten and ORL in terms of ACC, NMI, and ARI to evaluate the contribution of each component in CommonMVC. (1) The loss ablation contains three variants. Variant_1 denotes the erasure of the mutual information $I(X^v; Z^v)$. Variant_2 denotes the erasure of $D_{KL}(S_{i,\cdot}, S^v_{i,\cdot})$ and $D_{KL}(S_{\cdot,j}, S^v_{\cdot,j})$. Variant_3 denotes the erasure of the mutual information $I(Z; Z^v)$. From ablation results in Table 4, we can observe that the removal of each component of the loss function results in a degradation of performance. This observation underscores the necessity of each term in the loss function in ensuring optimal performance. Hence, these findings validate the rationality and efficacy of the loss function design. (2) The view ablation conducts view ablation on Handwritten and ORL datasets. As shown in Table 5, View_1 and View_2 represent the clustering results using only the first or second view, respectively. CommonMVC, utilizing both views with the nonlinear fusion strategy, outperforms individual views in all metrics. It shows the effectiveness of integrating

complementary information from multiple views. This demonstrates that the nonlinear fusion of multiple views in CommonMVC can capture more comprehensive patterns than relying on a single view.

Table 4. Loss analysis on the Handwritten and ORL dataset in terms of ACC, NMI and ARI.

Dataset	**Handwritten**			**ORL**		
Metric	***ACC***	***NMI***	***ARI***	***ACC***	***NMI***	***ARI***
Variant_1	0.722	0.728	0.621	0.538	0.653	0.403
Variant_2	0.725	0.709	0.614	0.630	0.745	0.497
Variant_3	0.909	0.831	0.810	0.658	0.771	0.524
CommonMVC	**0.912**	**0.848**	**0.823**	**0.711**	**0.848**	**0.580**

Table 5. View analysis on the Handwritten and ORL dataset in terms of ACC, NMI and ARI.

Dataset	**Handwritten**			**ORL**		
Metric	***ACC***	***NMI***	***ARI***	***ACC***	***NMI***	***ARI***
View_1	0.822	0.755	0.723	0.633	0.737	0.498
View_2	0.722	0.620	0.636	0.578	0.684	0.421
CommonMVC	**0.912**	**0.848**	**0.823**	**0.711**	**0.848**	**0.580**

4.4. Convergence Analysis

To validate the convergence behavior of CommonMVC, we analyzed the loss values across different training epochs on four distinct datasets, as depicted in Figure 3. The graph reveals a consistent and rapid decrease in the loss across all datasets, indicating that CommonMVC efficiently optimizes its objective functions during training. Notably, the convergence trend is evident early in the training process, with the loss stabilizing after approximately 60 epochs. The rapid convergence suggests that CommonMVC not only effectively captures the underlying data structure but also requires relatively few epochs to reach an optimal solution, making it a robust and efficient model for various datasets.

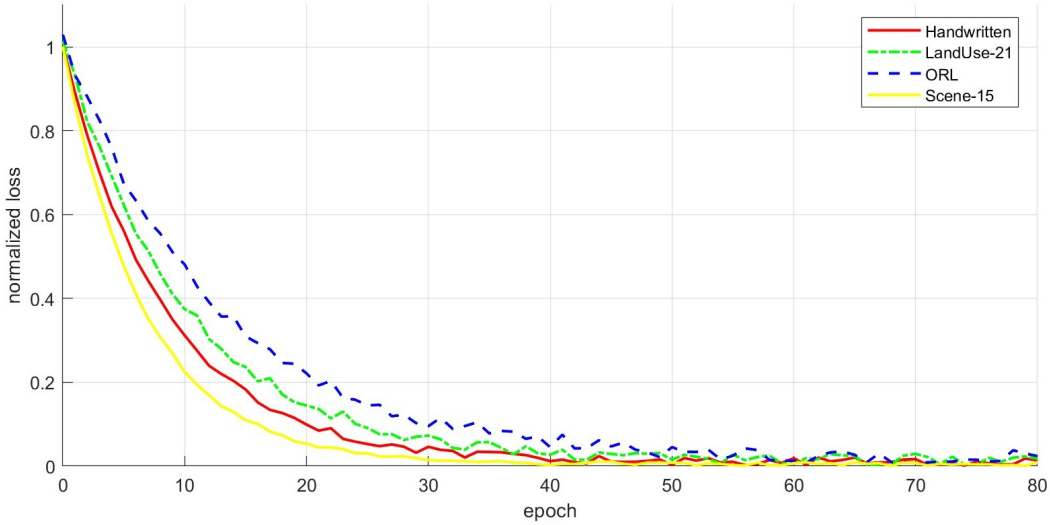
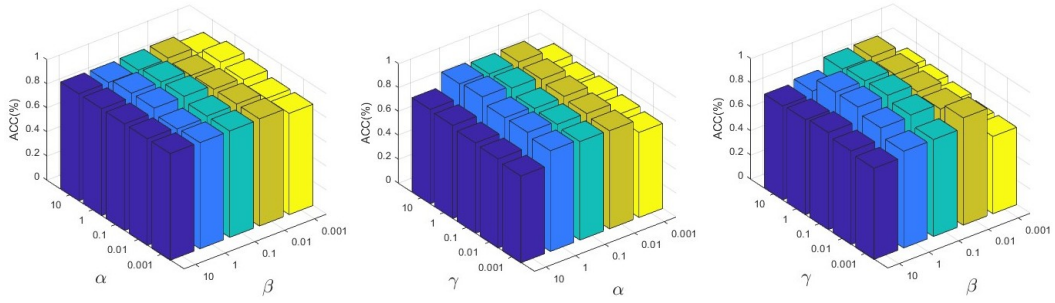


Figure 3. Convergence analysis on four datasets.

4.5. Parameter Analysis

In the hyperparameter analysis, we examine the impact of three key parameters α, γ, and β on the performance of CommonMVC. These parameters balance different terms in the overall loss function L of CommonMVC. We perform systematic hyperparameter

search experiments across the ranges of 10, 1, 0.1, 0.01, and 0.001 for all parameters. The experiments are carried out on the Handwritten dataset, with ACC as the performance metric. Figure 4 demonstrates the optimal result is gained when α is in the range of 0.1 to 1, and γ and β are within the range of 0.01 to 0.1. Under these hyperparameter settings, the model successfully balances common view learning with individual and common feature protection. This balance enabled the model to effectively extract cross-view common features while preserving individual characteristics, leading to the best ACC on the Handwritten dataset. The findings highlight the importance of selecting appropriate hyperparameters to achieve effective feature learning and clustering performance in CommonMVC.



(**a**) (**b**) (**c**)

Figure 4. Parameter analysis on the Handwritten dataset. (**a**) α v.s. β; (**b**) γ v.s. α; (**c**) γ v.s. β.

5. Conclusions

In this paper, we introduce CommonMVC, an innovative framework for multi-view clustering designed to mine patterns within multi-view data. CommonMVC disentangles and learns the complementary and consistent information hidden in multi-view data by defining view-specific representation learning and clustering partitioning components, along with nonlinear fusion functions. This approach uniquely designs a nonlinear fusion mechanism to enhance the complementary information between views and employs a relaxed semantic-level alignment strategy to ensure similar clustering predictions across different views, thereby improving the accuracy and robustness of the clustering outcomes. Comprehensive experiments conducted on four datasets have validated that CommonMVC achieves a new baseline for MVC tasks. CommonMVC outperforms existing baseline methods across various evaluation metrics. However, we acknowledge that our method has certain limitations. First, the computational complexity of the nonlinear fusion mechanism may increase with the number of views and the complexity of the data, which could limit its scalability for very large-scale multi-view datasets. Second, our current framework assumes that all views are available during training and testing. In scenarios where views may be missing, the performance of CommonMVC could be affected. Developing strategies to handle missing views and further optimizing the computational efficiency of the nonlinear fusion process will be important directions for future research.

Author Contributions: Methodology, Y.W.; Validation, H.L.; Investigation, H.L.; Data curation, H.L.; Writing—original draft, Y.W.; Writing—review & editing, Y.W. and H.L.; Supervision, H.L.; Funding acquisition, Y.W. and H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Qinghai Institute of Technology Doctoral Extension Program (2023011wys028), the Qinghai Kunlun Talented Person Project (2023-QLGKL YCZX-031), the Ministry of Education Humanities and Social Sciences Project (22XJC630006) and the Doctoral Foundation of Fuyang Normal University under Grant 2025KYQD0031.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Zhang, Y.; Cai, J.; Wu, Z.; Wang, P.; Ng, S.K. Mixture of Experts as Representation Learner for Deep Multi-View Clustering. *Proc. AAAI Conf. Artif. Intell.* **2025**, *39*, 22704–22713. [CrossRef]
2. Jiang, H.; Ma, W.; Dai, J.; Ding, J.; Tong, X.; Wang, Y.; Du, X.; Jiang, D.; Luo, Y.; Zhang, J. Cross representation subspace learning for multi-view clustering. *Expert Syst. Appl.* **2025**, *286*, 128007. [CrossRef]
3. Long, Z.; Wang, Q.; Ren, Y.; Liu, Y.; Zhu, C. TLRLF4MVC: Tensor Low-Rank and Low-Frequency for Scalable Multi-View Clustering. *IEEE Trans. Pattern Anal. Mach. Intell.* **2025**, *47*, 6900–6911. [CrossRef]
4. Bian, J.; Xie, X.; Wang, C.; Yang, L.; Lai, J.; Nie, F. Angular Reconstructive Discrete Embedding with Fusion Similarity for Multi-View Clustering. *IEEE Trans. Knowl. Data Eng.* **2025**, *37*, 45–59. [CrossRef]
5. Yu, Y.; Lu, Z.; Nie, F.; Yu, W.; Miao, Z.; Li, X. Pseudo-Label Guided Bidirectional Discriminative Deep Multi-View Subspace Clustering. *IEEE Trans. Knowl. Data Eng.* **2025**, *37*, 4213–4224. [CrossRef]
6. Zhang, P.; Pan, Y.; Wang, S.; Yu, S.; Xu, H.; Zhu, E.; Liu, X.; Tsang, I.W. Max-Mahalanobis Anchors Guidance for Multi-View Clustering. *Proc. AAAI Conf. Artif. Intell.* **2025**, *39*, 22488–22496. [CrossRef]
7. Yin, M.; Huang, W.; Gao, J. Shared Generative Latent Representation Learning for Multi-view Clustering. *Proc. AAAI Conf. Artif. Intell.* **2020**, *34*, 6688–6695. [CrossRef]
8. Cui, J.; Li, Y.; Fu, Y.; Wen, J. Multi-view Self-Expressive Subspace Clustering Network. In Proceedings of the 31st ACM International Conference on Multimedia 2023, Ottawa, ON, Canada, 29 October–3 November 2023; pp. 417–425.
9. Wan, Z.; Zhang, C.; Zhu, P.; Hu, Q. Multi-view Information-bottleneck Representation Learning. *Proc. AAAI Conf. Artif. Intell.* **2021**, *35*, 10085–10092. [CrossRef]
10. Peng, X.; Huang, Z.; Lv, J.; Zhu, H.; Zhou, J.T. COMIC: Multi-view Clustering Without Parameter Selection. In Proceedings of the International Conference on Machine Learning 2019, Long Beach, CA, USA, 9–15 June 2019; pp. 5092–5101.
11. Mao, Y.; Yan, X.; Guo, Q.; Ye, Y. Deep Mutual Information Maximin for Cross-modal Clustering. *Proc. AAAI Conf. Artif. Intell.* **2021**, *35*, 8893–8901. [CrossRef]
12. Xu, J.; Ren, Y.; Tang, H.; Yang, Z.; Pan, L.; Yang, Y.; Pu, X.; Yu, P.S.; He, L. Self-supervised Discriminative Feature Learning for Deep Multi-view Clustering. *IEEE Trans. Knowl. Data Eng.* **2022**, *35*, 7470–7482. [CrossRef]
13. Yang, X.; Jin, J.; Wang, S.; Liang, K.; Liu, Y.; Wen, Y.; Liu, S.; Zhou, S.; Liu, X.; Zhu, E. DEALMVC: Dual Contrastive Calibration for Multi-view Clustering. In Proceedings of the ACM International Conference on Multimedia 2023, Ottawa, ON, Canada, 29 October–3 November 2023; pp. 337–346.
14. Hu, S.; Tian, B.; Liu, W.; Ye, Y. Self-supervised Trusted Contrastive Multi-view Clustering with Uncertainty Refined. *Proc. AAAI Conf. Artif. Intell.* **2025**, *39*, 17305–17313. [CrossRef]
15. Wang, B.; Zeng, C.; Chen, M.; Li, X. Towards Learnable Anchor for Deep Multi-View Clustering. *Proc. AAAI Conf. Artif. Intell.* **2025**, *39*, 21044–21052. [CrossRef]
16. Fei, L.; He, J.; Zhu, Q.; Zhao, S.; Wen, J.; Xu, Y. Deep Multi-View Contrastive Clustering via Graph Structure Awareness. *IEEE Trans. Image Process.* **2025**, *34*, 3805–3816. [CrossRef] [PubMed]
17. Su, P.; Huang, S.; Ma, W.; Xiong, D.; Lv, J. Multi-view Granular-ball Contrastive Clustering. *Proc. AAAI Conf. Artif. Intell.* **2025**, *39*, 20637–20645. [CrossRef]
18. Gu, Z.; Feng, S. Individuality Meets Commonality: A Unified Graph Learning Framework for Multi-view Clustering. *ACM Trans. Knowl. Discov. Data* **2023**, *17*, 7. [CrossRef]
19. Chen, Z.; Wu, X.; Xu, T.; Li, H.; Kittler, J. Deep Discriminative Multi-View Clustering. *IEEE Trans. Circuits Syst. Video Technol.* **2025**, *35*, 6974–6978. [CrossRef]
20. Xu, J.; Meng, M.; Liu, J.; Wu, J. Deep multi-view clustering with diverse and discriminative feature learning. *Pattern Recognit.* **2025**, *161*, 111322. [CrossRef]
21. Gao, J.; Liu, M.; Li, P.; Zhang, J.; Chen, Z. Deep Multiview Adaptive Clustering with Semantic Invariance. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *35*, 12965–12978. [CrossRef]
22. Luong, K.; Nayak, R. A Novel Approach to Learning Consensus and Complementary Information for Multi-view Data Clustering. In Proceedings of the International Conference on Data Engineering 2020, Dallas, TX, USA, 20–24 April 2020; pp. 865–876.
23. Zheng, X.; Tang, C.; Liu, X.; Zhu, E. Multi-view Clustering via Matrix Factorization Assisted K-means. *Neurocomputing* **2023**, *534*, 45–54. [CrossRef]
24. Lin, Z.; Kang, Z. Graph Filter-based Multi-view Attributed Graph Clustering. In Proceedings of the International Joint Conference on Artificial Intelligence 2021, Montreal, QC, Canada, 19–27 August 2021; pp. 2723–2729.
25. Luo, S.; Zhang, C.; Zhang, W.; Cao, X. Consistent and Specific Multi-view Subspace Clustering. In Proceedings of the Conference on Artificial Intelligence 2018, New Orleans, LA, USA, 2–7 February 2018; pp. 3730–3737.

26. Cui, C.; Ren, Y.; Pu, J.; Pu, X.; He, L. Deep Multi-view Subspace Clustering with Anchor Graph. In Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence 2023, Macao, China, 19–25 August 2023; pp. 3577–3585.
27. Trosten, D.J.; Lokse, S.; Jenssen, R.; Kampffmeyer, M. Reconsidering Representation Alignment for Multi-view Clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2021, Nashville, TN, USA, 20–25 June 2021; pp. 1255–1265.
28. Gao, Q.; Lian, H.; Wang, Q.; Sun, G. Cross-modal Subspace Clustering via Deep Canonical Correlation Analysis. *Proc. AAAI Conf. Artif. Intell.* **2020**, *34*, 3938–3945. [CrossRef]
29. Zhang, C.; Liu, Y.; Fu, H. AE2-nets: Autoencoder in Autoencoder Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2019, Long Beach, CA, USA, 15–20 June 2019; pp. 2577–2585.
30. Wen, J.; Liu, C.; Xu, G.; Wu, Z.; Huang, C.; Fei, L.; Xu, Y. Highly Confident Local Structure Based Consensus Graph Learning for Incomplete Multi-view Clustering. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2023, Vancouver, BC, Canada, 17–24 June 2023; pp. 15712–15721.
31. Gao, J.; Liu, M.; Li, P.; Laghari, A.A.; Javed, A.R.; Victor, N.; Gadekallu, T.R. Deep Incomplete Multiview Clustering via Information Bottleneck for Pattern Mining of Data in Extreme-environment IoT. *IEEE Internet Things J.* **2023**, *11*, 26700–26712. [CrossRef]
32. Gao, J.; Li, P.; Laghari, A.A.; Srivastava, G.; Gadekallu, T.R.; Abbas, S.; Zhang, J. Incomplete Multiview Clustering via Semidiscrete Optimal Transport for Multimedia Data Mining in IoT. *ACM Trans. Multimed. Comput. Commun. Appl.* **2024**, *20*, 158. [CrossRef]
33. Li, P.; Laghari, A.A.; Rashid, M.; Gao, J.; Gadekallu, T.R.; Javed, A.R.; Yin, S. A Deep Multimodal Adversarial Cycle-consistent Network for Smart Enterprise System. *IEEE Trans. Ind. Inform.* **2022**, *19*, 693–702. [CrossRef]
34. Andrew, G.; Arora, R.; Bilmes, J.; Livescu, K. Deep Canonical Correlation Analysis. In Proceedings of the International Conference on Machine Learning 2013, Atlanta, GA, USA, 17–19 June 2013; pp. 1247–1255.
35. Wang, W.; Arora, R.; Livescu, K.; Bilmes, J. On Deep Multi-view Representation Learning. In Proceedings of the International Conference on Machine Learning 2015, Lille, France, 6–11 July 2015; pp. 1083–1092.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.