Electronic Research Archive

https://www.aimspress.com/journal/era

ERA, 32(9): 5424–5450.
DOI: 10.3934/era.2024252
Received: 25 June 2024
Revised: 13 September 2024
Accepted: 20 September 2024
Published: 26 September 2024

Research article

Deep Grassmannian multiview subspace clustering with contrastive learning

Rui Wang[1,2], **Haiqiang Li**[1], **Chen Hu**[1,2], **Xiao-Jun Wu**[1,2,*] **and Yingfang Bao**[3,4,*]

[1] School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi 214122, China
[2] Jiangsu Provincial Engineering Laboratory of Pattern Recognition and Computational Intelligence, Jiangnan University, Wuxi 214122, China
[3] Affiliated Wuxi Fifth Hospital of Jiangnan University, Wuxi 214007, China
[4] The Fifth People's Hospital of Wuxi, Wuxi 214007, China

* **Correspondence:** Email: wu_xiaojun@jiangnan.edu.cn, byf20090101@163.com.

Abstract: This paper investigated the problem of multiview subspace clustering, focusing on feature learning with submanifold structure and exploring the invariant representations of multiple views. A novel approach was proposed in this study, termed deep Grassmannian multiview subspace clustering with contrastive learning (DGMVCL). The proposed algorithm initially utilized a feature extraction module (FEM) to map the original input samples into a feature subspace. Subsequently, the manifold modeling module (MMM) was employed to map the aforementioned subspace features onto a Grassmannian manifold. Afterward, the designed Grassmannian manifold network was utilized for deep subspace learning. Finally, discriminative cluster assignments were achieved utilizing a contrastive learning mechanism. Extensive experiments conducted on five benchmarking datasets demonstrate the effectiveness of the proposed method. The source code is available at https://github.com/Zoo-LLi/DGMVCL.

Keywords: multiview clustering; contrastive learning; Grassmannian manifold; neural network; invariant representation

1. Introduction

Clustering stands as a foundational research area in the realm of computer vision and machine learning. Over decades of exploration, traditional clustering methods have achieved a certain level of maturity, yielding promising results. However, when confronted with complex signals (e.g., images and videos), clustering methods reliant on Euclidean distance measure often yield unsatisfactory performance. Although the data observed in practical applications is complicated, the clusters to which the samples belong are generally located near the corresponding low-dimensional space.

Therefore, some researchers have turned to subspace clustering to characterize the geometrical structure of the original data manifold. These algorithms transform each data point into multi-dimensional subspaces, fitting the data point to which it belongs through the low-dimensional subspace clusters to achieve the final clustering prediction. At present, subspace clustering methods catering to implicit subspace structures include iterative methods [1, 2], algebraic methods [3, 4], statistical methods [5, 6], and self-representation methods [7–9]. Among them, self-representation has received widespread attention due to their effective utilization of the latent subspace structure and attributes of the data. Notable representatives of such an approach include sparse subspace clustering (SSC) [7] and low-rank representation (LRR) [9]. These methods typically contain two steps: (1) exploring the data structure by using Euclidean distance measure and obtaining a coefficient matrix; (2) leveraging the learned coefficient matrix to conduct spectral clustering [10]. Then, the data is assigned to k clusters.

Nevertheless, previous studies reveal that image sets or video data often inhabit nonlinear manifold structure [11, 12]. Therefore, the Euclidean distance cannot accurately measure the similarity between any two data points residing on a non-Euclidean space. Wei et al. [13] proposed a discrete metric learning method for fast image set classification, which significantly improves classification efficiency and accuracy by learning metrics and hashing techniques on the Riemannian manifold. Furthermore, video data commonly feature varying frames per video clip, leading to exaggerated data dimensions upon the vectorization of video frames. These challenges pose difficulties for original clustering algorithms such as SSC or LRR.

Deep learning methods emerge as capable learners of discriminative feature representations [14]. A number of studies propose combining metric learning and deep learning for classification [14, 15]. Recently, deep learning techniques have started to be used in the scenario of data clustering and have shown remarkable performance [16–18]. To name a few, Xie et al. [19] proposed a clustering method based on deep embedding, where a deep neural network learns feature representations and clustering assignments concurrently. Wang et al. [20] proposed a multi-level representation learning method for incomplete multiview clustering, which incorporates contrastive and adversarial regularization to improve clustering performance. Yang et al. [21] proposed a novel graph contrastive learning framework for clustering multi-layer networks, effectively combining nonnegative matrix factorization and contrastive learning to enhance the discriminative features of vertices across different network layers. Guo et al. [22] proposed an adaptive multiview subspace learning method using distributed optimization, which enhances clustering performance by capturing high-order correlations among views. Li et al. [23] introduced a deep adversarial MVC method that explores the intrinsic structure embedded in multiview data. Ma et al. [24] suggested a deep generative clustering model based on variational autoencoders, being able to yield a more appropriate embedding subspace for clustering with respect to complex data distributions. These methods address challenges faced by traditional data modeling (e.g., preprocessing high-dimensional data and characterizing nonlinear relationships) by nonlinearly mapping data points into a latent space through a series of encoder layers. However, existing deep clustering algorithms have an inherent limitation, i.e., they utilize Euclidean computations to analyze the semantic features generated by convolutional neural networks, which leads to unfaithful data representation. The fundamental reason is that these features inherently have a submanifold structure [25, 26]. While techniques such as fine-tuning, optimization, and feature activation in deep neural networks can affect the experimental results, manifold learning provides a

theoretical basis for the analysis of non-Euclidean data structures.

Recognizing the nonlinear structure of high-dimensional data, some studies [11, 27–31] extended the traditional clustering paradigm to the context of Riemannian manifolds. Typically, a manifold represents a smooth surface embedded in Euclidean space [32]. In image analysis, covariance matrices often serve as region descriptors, with these matrices regarded as points on the symmetric positive definite (SPD) manifold. Hu et al. [12] proposed a multi-geometry SSC method, aiming to mine complementary geometric representations. Wei et al. [33] proposed a method called sparse representation classifier guided Grassmann reconstruction metric learning (GRML), which enhances image set classification by learning a robust metric on the Grassmann manifold, making it effective in handling noisy data. Linear subspaces have attracted widespread attention in many scientific fields, as their underlying space is a Grassmannian manifold, which provides a solid theoretical foundation for the characterization of signal data (e.g., video clips, image sets, and point clouds). In addition, the Grassmannian manifold plays a crucial role in handling non-Euclidean data structures, particularly through its compact manifold structure. Wang et al. [34] extended the ideology of LRR to the context of the Grassmannian manifold, facilitating more accurate representation and analysis in complicated data scenarios. Piao et al. [35] proposed a double-kernel norm low-rank representation based on the Grassmannian manifold for clustering. However, the issues such as integrating manifold representation learning with clustering and preserving subspace structure in the data tramsformation process are still challenging for such a framework.

It can be concluded that deep learning-based clustering struggles to learn effective representations from the data with a submanifold structure, and the underlying space of subspace features is a Grassmannian manifold. This motivates us to propose the framework of DGMVCL to achieve a more reasonable view-invariant representation on a non-Euclidean space. The proposed framework comprises four main components: the FEM, MMM, Grassmannian manifold learning module (GMLM), and the contrastive leaning module (CLM). Specifically, the FEM is responsible for mapping the original data into a feature subspace, the MMM is designed for the projection of subspace features onto a Grassmannian manifold, the GMLM is used to facilitate deep subspace learning on the Grassmannian manifold, and the CLM is focused on discriminative cluster assignments through contrastive learning. Additionally, the positive and negative samples relied upon in the contrastive learning process are constructed on the basis of Riemannian distance on the Grassmannian manifold, which can better reflect the geometric distribution of the data. Extensive experiments across five benchmarking datasets validate the effectiveness of the proposed DGMVCL.

Our main contributions are summarized as follows:

- A lightweight geometric learning model build upon the Grassmannian manifold is proposed for multiview subspace clustering in an end-to-end manner.
- A Grassmannian-level contrastive learning strategy is suggested to help improve the accuracy of cluster assignments among multiple views.
- Extensive experiments conducted on five multiview datasets demonstrate the effectiveness of the proposed DGMVCL.

2. Related works

In this section, we will give a brief introduction to some related works, including multiview subspace clustering, contrastive learning, and the Riemannian geometry of Grassmannian manifold.

2.1. Multiview subspace clustering

Although there exists a number of subspace clustering methods, such as LRR [9] and SSC [7], most of them adopt self-representation to obtain subspace features. Low-rank tensor-constrained multiview subspace clustering (LMSC) [36] is able to generate a common subspace for different views instead of individual representations. Flexible multiview representation learning for subspace clustering (FMR) [37] avoids using partial information for data reconstruction. Zhou et al. [38] proposed an end-to-end adversarial attention network to align latent feature distributions and assess the importance of different modalities. Despite the improved clustering performance, these methods do not consider the semantic label consistency across multiple views, potentially resulting in challenges when learning consistent clustering assignments.

To address these challenges, Kang et al. [39] proposed a structured graph learning method that constructs a bipartite graph to manage the relationships between samples and anchor points, effectively reducing computational complexity in large-scale data and enabling support for out-of-sample extensions. This approach is particularly advantageous in scalable subspace clustering scenarios, extending from single-view to multiview settings.

Furthermore, to enhance clustering performance by leveraging high-order structural information, Pan and Kang [40] introduced a high-order multiview clustering (HMvC) method. This approach utilizes graph filtering and an adaptive graph fusion mechanism to explore long-distance interactions between different views. By capturing high-order neighborhood relationships, HMvC effectively improves clustering results on both graph and non-graph data, addressing some of the limitations in prior methods that overlooked the intrinsic high-order information from data attributes.

2.2. Contrastive learning

Contrastive learning has made significant progress in self-supervised learning [41–44]. The methods based on contrastive learning essentially rely on a large number of pairwise feature comparisons. Specifically, they aim to maximize the similarity between positive samples in the latent feature space while minimizing the similarity between negative samples simultaneously. In the field of clustering, positive samples are constructed from the invariant representations of all multiview instances of the same sample, while negative samples are obtained by pairing representations from different samples across various views. Chen et al. [42] proposed a contrastive learning framework that maximizes consistency between differently augmented views of the same example in the latent feature space. Wang et al. [44] investigated two key properties of the contrastive loss, namely feature alignment from positive samples and uniformity of induced feature distribution on the hypersphere, which can be used to measure the quality of generated samples. Although these methods can learn robust representations based on data augmentation, learning invariant representations across multiple views remains challenging.

2.3. Grassmannian manifold

The Grassmannian manifold $\mathcal{G}(q, d)$ comprises a collection of q-dimensional linear subspaces within $\mathbb{R}^d$. Each of them can be naturally represented by an orthonormal basis denoted as $\mathbf{Y}$, with the size of $d \times q$ ($\mathbf{Y}^{\mathrm{T}}\mathbf{Y} = \mathbf{I}_q$, where $\mathbf{I}_q$ is a $q \times q$ identity matrix). Consequently, the matrix representation of each Grassmannian element is comprised of an equivalence class of this orthonormal basis:

$$[\mathbf{Y}] = \{\widetilde{\mathbf{Y}} \mid \widetilde{\mathbf{Y}} = \mathbf{YO}, \mathbf{O} \in \mathrm{O}(q)\}, \tag{2.1}$$

where $\mathbf{Y}$ represents a $d \times q$ column-wise orthonormal matrix. The definition in Eq (2.1) is commonly referred to as the orthonormal basis (ONB) viewpoint [32].

As demonstrated in [32], each point of the Grassmannian manifold can be alternatively represented as an idempotent symmetric matrix of rank q, given by $\Phi(\mathbf{Y}) = \mathbf{YY}^{\mathrm{T}}$. This representation, known as the projector perspective (PP), signifies that the Grassmannian manifold is a submanifold of the Euclidean space of symmetric matrices. Therefore, an extrinsic distance can be induced by the ambient Euclidean space, termed the projection metric (PM) [49]. The PM is defined as:

$$\mathrm{d}_{\mathrm{PM}}(\mathbf{Y}_1, \mathbf{Y}_2) = 2^{-1/2}\|\mathbf{Y}_1\mathbf{Y}_1^{\mathrm{T}} - \mathbf{Y}_2\mathbf{Y}_2^{\mathrm{T}}\|_{\mathrm{F}}, \tag{2.2}$$

where $\|\cdot\|_{\mathrm{F}}$ denotes the Frobenius norm. As evidenced in [45], the distance computed by the PM deviates from the true geodesic distance on the Grassmannian manifold by a scale factor of $\sqrt{2}$, making it a widely used Grassmannian distance.

3. Proposed method

3.1. Network structure

We are given a group of multiview data $\mathbf{X} = \{\mathbf{X}^v \in \mathbb{R}^{d_v \times N_v}\}_{v=1}^{V}$, where V denotes the number of views, N_v signifies the number of instances contained in the v-th view, and d_v represents the dimensionality of each instance in $\mathbf{X}^v$. The proposed DGMVCL is an end-to-end neural network built upon the Grassmannian manifold, aiming to generate clustering semantic labels from the original instances across multiple views. As shown in Figure 1, the proposed framework mainly consists of four modules, which are the FEM, MMM, GMLM, and CLM, respectively. The following is a detailed introduction to them.

3.1.1. FEM

To obtain an effective semantic space for the subsequent computations, we exploit a CNN network to transform the input images into subspace representations with lower redundancy. The proposed FEM contains two blocks, each of which comprises a convolutional layer, a ReLU activation layer, and a max-pooling layer, respectively. The difference of the two blocks lies in the number of convolutional kernels involved, i.e., 32 for the first convolutional layer and 64 for the second one. To characterize the geometric structure of the generated features, the following manifold modeling module is designed.

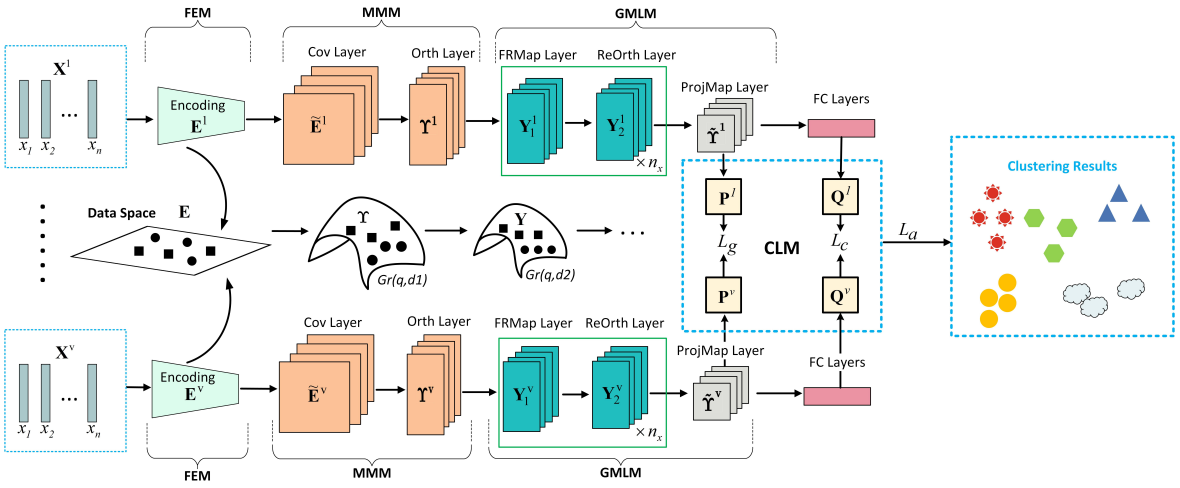


Figure 1. An overview of the proposed DGMVCL. It can be seen that this framework is made up of four different modules, i.e., FEM, MMM, GMLM, and CLM, respectively. Besides, n_x represents the number of Grassmannian operation blocks contained in GMLM.

3.1.2. MMM

Let $\mathbf{E}_i^v \in \mathbb{R}^{c\times l}$ be the i-th ($i \in \{1, 2, ..., N_v\}$) feature matrix generated by FEM with respect to the i-th input instance of the v-th view. Here, c represents the number of channels while l indicates the length of a vectorized feature map. Since each point of $\mathcal{G}(q, d)$ represents a q-dimentional linear subspace of the d-dimentional vector space $\mathbb{R}^d$ (see Section 2.3), the Grassmannian manifold thus becomes a reasonable and efficient tool for parametrizing the q-dimentional real vector subspace embedded in $\mathbf{E}_i^v$ [49, 51].

Cov Layer: To capture complementary statistical information embodied in different channel features, a similarity matrix is computed for each $\mathbf{E}_i^v$:

$$\tilde{\mathbf{E}}_i^v = \mathbf{E}_i^v(\mathbf{E}_i^v)^{\mathbf{T}}. \tag{3.1}$$

Orth Layer: Following the Cov layer, the SVD operation is applied to obtain a q-dimensional linear subspace spanned by an orthonormal matrix $\mathbf{Y}_i^v \in \mathbb{R}^{c\times q}$, that is $\tilde{\mathbf{E}}_i^v \simeq \mathbf{Y}_i^v\mathbf{\Sigma}_i^v(\mathbf{Y}_i^v)^{\mathrm{T}}$, wherein, $\mathbf{Y}_i^v$ and $\mathbf{\Sigma}_i^v$ are two matrices consisting of q leading eigenvalues and the corresponding eigenvectors, respectively. Now, the resulting Grassmannian representation with respect to the input $\mathbf{X}^v$ is denoted by $\mathbf{\Upsilon}^v = [\mathbf{Y}_1^v, \mathbf{Y}_2^v, ..., \mathbf{Y}_{N_v}^v]$.

3.1.3. GMLM

An overview of the designed GMLM is shown in Figure 1. We can note that the input to this module is a series of orthonormal matrices. For simplicity, we abbreviate $\mathbf{Y}_i^v$ as $\mathbf{Y}_i$ in the following. Besides, since $\mathbf{Y}_i \in \mathcal{G}(q, c)$, we replace the symbol c with the symbol d. To implement deep subspace learning, the following three basic layers are designed.

FRMap Layer: This layer transforms the input orthogonal matrices into new ones through a linear mapping function f_{fr}:

$$\mathbf{Y}_{i,k} = f_{fr}^{(k)}(\mathbf{Y}_{i,k-1}; \mathbf{W}_k) = \mathbf{W}_k \mathbf{Y}_{i,k-1}, \tag{3.2}$$

where $\mathbf{Y}_{i,k-1} \in \mathcal{G}(q, d_{k-1})$ is the input data of the k-th layer, $\mathbf{W}_k \in \mathbb{R}^{d_k \times d_{k-1}}$ $(d_k < d_{k-1})$ is the to-be-learned transformation matrix (connection weights), essentially required to be a row full-rank matrix, and $\mathbf{Y}_{i,k} \in \mathbb{R}^{d_k \times q}$ is the resulting matrix. Due to the fact that the weight matrices lie in a non-compact Stiefel manifold and the geodesic distance has no upper bound [46, 47], direct optimization on such a manifold is unfeasible. To address this challenge, we follow [46] to impose an orthogonality constraint on each weight matrix $\mathbf{W}_k$. As a consequence, the weight space $\mathbb{R}_*^{d_k \times d_{k-1}}$ becomes a compact Stiefel manifold $St(d_k, d_{k-1})$ [48], facilitating better optimization.

ReOrth Layer: Inspired by [46], we design the ReOrth layer for the sake of preventing the output matrices of the FRMap layer from degeneracy. Specifically, we first impose QR decomposition on the input matrix $\mathbf{Y}_{i,k-1}$:

$$\mathbf{Y}_{i,k-1} = \mathbf{Q}_{i,k-1} \mathbf{R}_{i,k-1}, \tag{3.3}$$

where $\mathbf{Q}_{i,k-1} \in \mathbb{R}^{d_{k-1} \times q}$ is an orthogonal matrix, and $\mathbf{R}_{i,k-1} \in \mathbb{R}^{q \times q}$ is an invertible upper triangular matrix. Therefore, $\mathbf{Y}_{i,k-1}$ can be normalized into an orthonormal basis matrix via the following transformation function $f_{ro}^{(k)}$:

$$\mathbf{Y}_{i,k} = f_{ro}^{(k)}(\mathbf{Y}_{i,k-1}) = \mathbf{Y}_{i,k-1} \mathbf{R}_{i,k-1}^{-1} = \mathbf{Q}_{i,k-1}. \tag{3.4}$$

ProjMap Layer: As studied in [47, 49–52], the PM is one of the most popular Grassmannian distance measures, providing a specific inner product structure to a concrete Riemannian manifold. In such a case, the original Grassmannian manifold reduces to a flat space, in which the Euclidean computations can be generalized to the projection domain of orthogonal matrices. Formally, by applying the projection operator [47] to the orthogonal matrix $\mathbf{Y}_{i,k-1}$ of the k-th layer, the designed ProjMap layer can be formulated as:

$$\mathbf{Y}_{i,k} = f_{pm}^{(k)}(\mathbf{Y}_{i,k-1}) = \mathbf{Y}_{i,k-1} \mathbf{Y}_{i,k-1}^{\mathbf{T}}. \tag{3.5}$$

Subsequently, we embed a contrastive learning module at the end of the GMLM to enhance the discriminatory power of the learned features.

3.1.4. CLM

For simplicity, we assume that $n_x = 1$. In this case, the output data representation with respect to $\mathbf{\Upsilon}^v$ is denoted by $\tilde{\mathbf{\Upsilon}}^v = [\mathbf{Y}_{1,3}^v, \mathbf{Y}_{2,3}^v, ..., \mathbf{Y}_{N_v,3}^v]$. Then, the projection metric, defined in Eq (2.2), is applied to compute the geodesic distance between any two data points, enabling the execution of K-means clustering within each view. At the same time, we can obtain the membership degree t_{ij}, computed as follows:

$$t_{ij} = \frac{\frac{1}{d_{\mathrm{PM}}^2(\mathbf{Y}_{i,3}^v, \mathbf{U}_j)}}{\sum_{r=1}^{K} \frac{1}{d_{\mathrm{PM}}^2(\mathbf{Y}_{i,3}^v, \mathbf{U}_r)}}, \tag{3.6}$$

where K represents the total number of clusters, and $\mathbf{U}_j$ denotes the j-th ($j \in \{1, 2, ..., K\}$) cluster. To enhance the discriminability of the global soft assignments, we consider a unified target distribution probability $\mathbf{P}^v \in \mathbb{R}^{N_v \times K}$, which is formulated as [53]:

$$p_{ij}^v = \frac{\left(t_{ij}^2 / \sum_i^{N_v} t_{ij}\right)}{\sum_j^K \left(t_{ij}^2 / \sum_i^{N_v} t_{ij}\right)}, \tag{3.7}$$

where each p_{ij}^v represents the soft cluster assignment of the i-th sample to the j-th cluster. Therefore, $\mathbf{p}_j^v$ represents the cluster assignments of the same semantic cluster.

The similarity between the two cluster assignments $\mathbf{p}_j^{v_1}$ and $\mathbf{p}_j^{v_2}$ for cluster j is measured by the following cosine similarity [54]:

$$s(\mathbf{p}_j^{v_1}, \mathbf{p}_j^{v_2}) = \frac{\mathbf{p}_j^{v_1} \cdot \mathbf{p}_j^{v_2}}{\|\mathbf{p}_j^{v_1}\| \cdot \|\mathbf{p}_j^{v_2}\|}, \tag{3.8}$$

where v_1 and v_2 represent two different views. Since these instances belong to the same labels in each view, the cluster assignment probabilities of instances from different views should be similar. Moreover, instances from multiple views are independent of each other for different samples. Therefore, for V views with K clusters, there are $(V - 1)$ positive cluster assignment pairs and $V(K - 1)$ negative cluster assignment pairs.

The goal of CLM is to maximize the similarity between cluster assignments within clusters and minimize the similarity between cluster assignments across clusters. Inspired by [55], the cross-view contrastive loss between $\mathbf{p}_k^{v_1}$ and $\mathbf{p}_k^{v_2}$ is defined as follows:

$$l^{(v1,v2)} = -\frac{1}{K} \sum_{k=1}^{K} \log \frac{e^{s(\mathbf{p}_k^{v_1}, \mathbf{p}_k^{v_2})/\tau}}{\sum_{j=1, j\neq k}^{K} e^{s(\mathbf{p}_j^{v_1}, \mathbf{p}_k^{v_1})/\tau} + \sum_{j=1}^{K} e^{s(\mathbf{p}_j^{v_1}, \mathbf{p}_k^{v_2})/\tau}}, \tag{3.9}$$

where τ is the temperature parameter, $(\mathbf{p}_k^{v_1}, \mathbf{p}_k^{v_2})$ is a positive cluster assignment pair between views v_1 and v_2, and $(\mathbf{p}_j^{v_1}, \mathbf{p}_k^{v_1})$ $(j \neq k)$, $(\mathbf{p}_j^{v_1}, \mathbf{p}_k^{v_2})$ are negative cluster assignment pairs in the views of v_1 and v_2, respectively. The introduced cross-view contrastive loss across multiple views is given below:

$$L_g = \frac{1}{2} \sum_{v_1=1}^{V} \sum_{v_2=1, v_2\neq v_1}^{V} l^{(v_1,v_2)}. \tag{3.10}$$

The cross-view contrastive loss explicitly pulls together cluster assignments within the same cluster and pushes apart cluster assignment pairs from different clusters. This inspiration comes from the recently proposed contrastive learning paradigm, which is applied to semantic labels to explore consistent information across multiple views.

Additionally, to ensure consistency between cluster labels on the Grassmannian manifold and Euclidean space, as shown in Figure 1, we append two linear layers and a softmax function to the tail of the ProjMap layer to generate a probability matrix $\mathbf{Q}^v \in \mathbb{R}^{N_v \times K}$ for cluster assignments. Let $\mathbf{q}_i^v$ be the i-th row in $\mathbf{Q}^v$, and q_{ij}^v represents the probability that the i-th instance belongs to the j-th cluster of the v-th view. The semantic label of the i-th instance is determined by the maximum value in $\mathbf{q}_i^v$. Following similar steps as processing L_g, we get the contrastive loss L_c for different views in

Euclidean space:

$$L_c = -\frac{1}{2K}\sum_{v_1=1}^{V}\sum_{v_2=1,v_2\neq v_1}^{V}\sum_{k=1}^{K}\log\frac{e^{s(\mathbf{q}_k^{v_1},\mathbf{q}_k^{v_2})/\tau}}{\sum_{j=1,j\neq k}^{K}e^{s(\mathbf{q}_j^{v_1},\mathbf{q}_k^{v_1})/\tau}+\sum_{j=1}^{K}e^{s(\mathbf{q}_j^{v_1},\mathbf{q}_k^{v_2})/\tau}}. \tag{3.11}$$

Inspired by [53], we introduce the following regularization term to prevent all the instances from being assigned to a specific cluster:

$$L_a = \sum_{v=1}^{V}\sum_{j=1}^{K}\mathbf{h}_j^v\log\mathbf{h}_j^v, \tag{3.12}$$

where $\mathbf{h}_j^v = \frac{1}{N_v}\sum_{i=1}^{N_v}q_{ij}^v$. This term is considered as the cross-view consistency loss in DGMVCL.

3.1.5. Label prediction

The objective function of the proposed method comprises three primary components: the Grassmannian contrastive loss, the Euclidean contrastive loss, and the cross-view consistency loss, given below:

$$L_{\text{obj}} = \alpha L_g + \beta L_c + \gamma L_a, \tag{3.13}$$

where α, β, and γ are three trade-off parameters.

The goal of L_{obj} is to learn common semantic labels from feature representations in multiple views. Let $\mathbf{p}_i^v$ be the i-th row of $\mathbf{P}^v$, and p_{ij}^v represents the j-th element of $\mathbf{p}_i^v$. Specifically, $\mathbf{p}_i^v$ is a K-dimensional soft assignment probability, where $\sum_{j=1}^{K}p_{ij}^v = 1$. Once the training process of the proposed network is completed, the semantic label of sample i ($i \in 1, 2, ..., N_v$) can be predicted by:

$$y_i = \arg\max_j\left(\frac{1}{V}\sum_{v=1}^{V}p_{ij}^v\right). \tag{3.14}$$

3.2. Optimization

For the proposed GMLM, the composition of a series of successive functions $f = f^{(\rho)}\circ f^{(\rho-1)}\circ f^{(\rho-2)}\circ ...\circ f^{(2)}\circ f^{(1)}$ with $\boldsymbol{W} = \{\mathbf{W}_\rho, \mathbf{W}_{\rho-1}, ..., \mathbf{W}_1\}$ is the parameter tuple which can be viewed as the data embedding model, which satisfies the properties of metric space. Here, $f^{(k)}$ and $\mathbf{W}_k$ are, respectively, the operation function and weight parameter of the k-th layer, and ρ denotes the number of layers contained in GMLM. The loss of the k-th layer can be signified as: $L^{(k)} = \ell\circ f^{(\rho)}\circ ...\circ f^{(k)}$, where ℓ is actually the L_{obj}.

Due to the fact that the weight space of the FRMap layer is a compact Stiefel manifold $St(d_k, d_{k-1})$, we refer to the method studied in [46] to realize parameter optimization by generalizing the traditional stochastic gradient descent (SGD) settings to the context of Stiefel manifolds. The updating rule for $\mathbf{W}_k$ is given below:

According to Eq (3.2), we can have that: $\mathbf{Y}_k = f^{(k)}(\mathbf{W}_k, \mathbf{Y}_{k-1}) = \mathbf{W}_k\mathbf{Y}_{k-1}$. Then, the following variation of $\mathbf{Y}_k$ can be derived:

$$d\mathbf{Y}_k = d\mathbf{W}_k\mathbf{Y}_{k-1} + \mathbf{W}_k d\mathbf{Y}_{k-1}. \tag{3.15}$$

Based on the invariance of the first-order differential, the following chain rule can be deduced:

$$\frac{\partial L^{(k+1)}}{\partial \mathbf{Y}_k} : d\mathbf{Y}_k = \frac{\partial L^{(k)}}{\partial \mathbf{W}_k} : d\mathbf{W}_k + \frac{\partial L^{(k)}}{\partial \mathbf{Y}_{k-1}} : d\mathbf{Y}_{k-1}. \tag{3.16}$$

By replacing the left-hand side of Eq (3.16) with Eq (3.15) and exploiting the matrix inner product ":" property, the following two formulas can be derived:

$$\frac{\partial L^{(k+1)}}{\partial \mathbf{Y}_k} : d\mathbf{W}_k\mathbf{Y}_{k-1} = \frac{\partial L^{(k+1)}}{\partial \mathbf{Y}_k}\mathbf{Y}_{k-1}^{\mathrm{T}} : d\mathbf{W}_k, \tag{3.17}$$

$$\frac{\partial L^{(k+1)}}{\partial \mathbf{Y}_k} : \mathbf{W}_k d\mathbf{Y}_{k-1} = \mathbf{W}_k^{\mathrm{T}}\frac{\partial L^{(k+1)}}{\partial \mathbf{Y}_k} : d\mathbf{Y}_{k-1}. \tag{3.18}$$

Combining Eqs (3.16)–(3.18), the partial derivatives of $L^{(k)}$ with respect to $\mathbf{W}_k$ and $\mathbf{Y}_{k-1}$ can be computed by:

$$\frac{\partial L^{(k)}}{\partial \mathbf{W}_k} = \frac{\partial L^{(k+1)}}{\partial \mathbf{Y}_k}\mathbf{Y}_{k-1}^{\mathrm{T}}, \quad \frac{\partial L^{(k)}}{\partial \mathbf{Y}_{k-1}} = \mathbf{W}_k^{\mathrm{T}}\frac{\partial L^{(k+1)}}{\partial \mathbf{Y}_k}. \tag{3.19}$$

At this time, the updating criterion of $\boldsymbol{W}_k$ on the Stiefel manifold is given below:

$$\mathbf{W}_k^{t+1} = \mathcal{R}_{\mathbf{W}_k^t}(-\eta\Pi_{\mathbf{W}_k^t}(\nabla L^{(k)}_{\boldsymbol{W}_k^t})), \tag{3.20}$$

where $\mathcal{R}$ signifies the retraction operation used to map the optimized parameter back onto the Stiefel manifold, η is the learning rate, and Π represents the projection operator used to convert the Euclidean gradient into the corresponding Riemannian counterpart:

$$\tilde{\nabla} L^{(k)}_{\boldsymbol{W}_k^t} = \Pi_{\mathbf{W}_k^t}(\nabla L^{(k)}_{\boldsymbol{W}_k^t}) = \nabla L^{(k)}_{\boldsymbol{W}_k^t} - \boldsymbol{W}_k^t(\nabla L^{(k)}_{\boldsymbol{W}_k^t})^{\mathrm{T}}\boldsymbol{W}_k^t, \tag{3.21}$$

where $\nabla L^{(k)}_{\boldsymbol{W}_k^t}$ is the Euclidean gradient, computed by the first term of Eq (3.19), and $\tilde{\nabla} L^{(k)}_{\boldsymbol{W}_k^t}$ denotes the Riemannian gradient. After that, the weight parameter can be updated by: $\boldsymbol{W}_k^{t+1} = \mathcal{R}(\boldsymbol{W}_k^t - \eta\tilde{\nabla} L^{(k)}_{\boldsymbol{W}_k^t})$. For detailed information about the Riemannian geometry of a Stiefel manifold and its associated retraction operation, please kindly refer to [48].

4. Experiments

In this section, we conduct experiments on five benchmarking datasets to evaluate the performance of the proposed DGMVCL. All the experiments are run on a Linux workstation with a GeForce RTX 4070 GPU (12 GB caches).

4.1. Experimental settings

4.1.1. Datasets

The proposed DGMVCL is evaluated on five publicly available multiview datasets. The MNIST-USPS [56] dataset contains 5000 samples with two different styles of digital images. The Multi-COIL-10 dataset [57] is comprised of 720 grayscale images collected from 10 categories with the image size

of 32×32, where different views represent different object poses. The Fashion dataset [58] is made up of 10,000 images belonging to 10 categories, where three different styles of an object are regarded as its three different views, i.e., front view, side view, and back view, respectively. The ORL dataset [59] consists of 400 face images collected from 40 volunteers, with each volunteer providing 10 images under different expressions and lighting conditions. The Scene-15 [60] dataset contains 4485 scene images belonging to 15 categories.

4.1.2. Evaluation metrics

We evaluate the clustering performance using the following three metrics, i.e., clustering accuracy (ACC), normalized mutual information (NMI), and purity. Here, ACC is the ratio of correctly classified samples to the total number of samples, NMI is an indicator to measure the consistency between the clustering result and the true class distribution, and purity refers to the ratio of the number of samples in the largest cluster to the total number of samples, indicating whether each cluster contains samples belonging to the same class.

Table 1. Results of all methods on the MNIST-USPS, Fashion, and Multi-COIL-10 datasets.

Datasets	Methods	ACC	NMI	Purity
MNIST-USPS	BSVC [61]	67.98	74.43	72.34
	SC_{Agg} [62]	89.00	77.12	89.18
	ASR [41]	97.90	94.72	97.90
	DSIMVC [63]	99.34	98.13	99.34
	DCP [64]	99.02	97.29	99.02
	MFL [65]	99.66	99.01	99.66
	CVCL [55]	99.58	98.79	99.58
	DGMVCL	**99.82**	**99.52**	**99.82**
Fashion	BSVC [61]	60.32	64.91	63.84
	SC_{Agg} [62]	98.00	94.80	97.56
	ASR [41]	96.52	93.04	96.52
	DSIMVC [63]	88.21	83.99	88.21
	DCP [64]	89.37	88.61	89.37
	MFL [65]	99.20	98.00	99.20
	CVCL [55]	99.31	98.21	99.31
	DGMVCL	**99.52**	**98.73**	**99.52**
Multi-COIL-10	BSVC [61]	73.32	76.91	74.11
	SC_{Agg} [62]	68.34	70.18	69.26
	ASR [41]	84.23	65.47	84.23
	DSIMVC [63]	99.38	98.85	99.38
	DCP [64]	70.14	81.9	70.14
	MFL [65]	99.20	98.00	99.20
	CVCL [55]	99.43	99.04	99.43
	DGMVCL	**100.00**	**100.00**	**100.00**

Table 2. Results of all methods on the ORL and Scene-15 datasets.

Datasets	Methods	ACC	NMI	Purity
ORL	BSVC	61.31	64.91	61.31
	SC_{Agg} [62]	61.65	77.41	66.22
	ASR [41]	79.49	78.04	81.49
	DSIMVC [63]	25.37	52.91	25.37
	DCP [64]	27.70	49.93	27.70
	MFL [65]	80.03	89.34	80.03
	CVCL [55]	85.50	93.17	86.00
	DGMVCL	**92.25**	**98.34**	**92.25**
Scene-15	BSVC	38.05	38.85	42.08
	SC_{Agg} [62]	38.13	39.31	44.76
	ASR [41]	42.70	40.70	45.60
	DSIMVC [63]	28.27	29.04	29.79
	DCP [64]	42.32	40.38	43.85
	MFL [65]	42.52	40.34	44.53
	CVCL [55]	44.59	42.17	47.36
	DGMVCL	**61.29**	**76.39**	**65.04**

4.2. Comparative methods

To validate the effectiveness of the proposed method, we compare DGMVCL with several state-of-the-art methods, including augmented sparse representation (ASR) [41], deep save IMVC (DSIMVC) [63], dual contrastive prediction (DCP) [64], multi-level feature learning (MFL) [65], and cross-view contrastive learning (CVCL) [55]. For DCP, we report the best clustering results obtained for each pair of individual views in each dataset. For better comparison, we include two additional baselines. Specifically, we first apply spectral clustering [61] to each individual view and report the best clustering results obtained across multiple views, termed as best single view clustering (BSVC). Then, we utilize an adaptive neighborhood graph learning method [62] to generate a similarity matrix for each individual view. We aggregate all the similarity matrices into a new one for spectral clustering, denoted as SC_{Agg}.

4.3. Performance evaluation

The clustering results of various algorithms obtained on the five used datasets are reported in Tables 1 and 2, respectively. The best results are highlighted in bold. It can be seen that the clustering performance of the contrastive learning-based methods, including DGMVCL, CVCL, MFL, and DCP, are superior to other competitors (e.g., BSVC and SC_{Agg}) on the large-scale MNIST-USPS, Scene-15, and Fashion datasets. This is mainly attributed to the fact that the contrastive learning-based self-supervised learning mechanism is capable of learning more discriminative feature representations by maximizing the similarity between positive samples and minimizing the similarity between negative samples simultaneously. Furthermore, our proposed DGMVCL is the best performer on all the used datasets, demonstrating its effectiveness. To name a few, on the Scene-15 dataset, the

DGMVCL achieves approximately 16.7%, 34.22%, and 17.68% improvements in ACC, NMI, and purity in comparison with the second-best CVCL method. The fundamental reason is that the proposed subspace-based geometric learning method can characterize and analyze the structural information of the input subspace features more faithfully. In addition, the introduced dual contrastive losses makes it possible to learn a more discriminative network embedding.

4.4. Ablation studies

Objective Function: To verify the impact of each loss function in Eq (3.13) on the performance of the proposed method, we conduct ablation experiments on the MNIST-USPS, ORL, and Fashion datasets as three examples. Table 3 shows the experimental results of our model under different combinations of loss functions. It can be seen that under type D, that is, all the loss functions are used at the same time, our method achieves the best performance. However, when L_a is removed (type A), the clustering performance of DGMVCL decreases on the three used datasets. For the MVC task, similar samples usually exhibit a large intra-data diversity, while dissimilar samples usually demonstrate a large inter-data correlation. This makes it impossible for a single L_c to effectively learn invariant representations from such data variations. Therefore, L_a is introduced to prevent instances from being assigned to a particular cluster, and the experimental results confirm its effectiveness. From Table 3, we can also conclude that the introduced Grassmannian contrastive loss L_g is beneficial for ameliorating the discriminability of the learned geometric features. Another interesting observation from Table 3 is that the clustering performance of type C (L_c is removed) is visibly lower than type D in terms of ACC, NMI, and purity. The fundamental reason is that L_c is the most critical loss function, as it is not only used to train the proposed network, but also participates in the testing phase. This indicates that L_c is crucial for distinguishing positive and negative samples, i.e., maintaining the cross-view consistency, and plays a pivotal role in the overall clustering task. Since L_g and L_a act as two regularization terms to proivide additional discriminative information for L_c and are mainly confined to the training phase, neither L_a nor L_g can be used alone. All in all, the aforementioned experimental findings demonstrate that each term in Eq (3.13) is useful.

Table 3. Comparison under different combinations of loss functions on the MNIST-USPS, ORL, and Fashion datasets.

	Loss			MNIST-USPS			ORL			Fashion		
Types	L_g	L_c	L_a	ACC	NMI	Purity	ACC	NMI	Purity	ACC	NMI	Purity
A	✓	✓		19.96	42.62	19.96	69.00	92.67	70.00	84.78	93.13	89.38
B		✓	✓	99.76	99.37	99.76	85.00	86.58	85.00	99.48	98.60	99.48
C	✓		✓	10.31	12.54	10.31	12.50	26.69	12.50	25.75	38.72	25.75
D	✓	✓	✓	99.82	99.52	99.82	92.25	98.34	92.25	99.52	98.73	99.52
E	✓			N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
F			✓	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A

The Components of GMLM: To verify the significance of each operation layer defined in GMLM, we make ablation experiments on the MNIST-USPS, Fashion, and Multi-COIL-10 datasets as three examples. The clustering results of different subarchitectures are listed in Table 4. It is evident that

group A (served as the reference group, with only the loss function L_g removed) is superior to group D in terms of ACC, NMI, and purity. Wherein, group D is generated by removing the ProjMap layer from group A. This demonstrates the necessity of Riemannian computation in preserving the geometric structure of the original data manifold. From Table 4, we can also find that the performance of group C (obtained by removing the layers of FRMap, ReOrth, and ProjMap from group A) is significantly inferior to that of group D on the Fashion and Multi-COIL-10 datasets, suggesting that deep subspace learning is able to improve the effectiveness of the features. Another interesting observation from Table 4 is that after expurgating the ReOrth layer from group A (this becomes group B), the clustering results are reduced to a certain extent on the three used datasets. The basic reason is that the Grassmannian properties, e.g., orthogonality, of the input feature matrices cannot be maintained, resulting in imprecise Riemannian distances computed in L_g. All in all, the experimental results mentioned above confirm the usefulness of each part in GMLM.

Table 4. Comparison under different subarchitectures of the proposed model on the MNIST-USPS, Fashion, and Multi-COIL-10 datasets.

	MNIST-USPS			Fashion			Multi-COIL-10		
Groups	ACC	NMI	Purity	ACC	NMI	Purity	ACC	NMI	Purity
A	99.76	99.37	99.76	99.48	98.60	99.48	100.00	100.00	100.00
B	99.70	99.11	99.70	99.02	97.60	99.02	99.14	98.71	99.14
C	98.00	97.00	97.00	81.25	86.24	81.40	31.70	31.00	32.00
D	89.92	94.51	89.92	98.49	96.52	98.49	99.00	97.31	99.01

Table 5. Comparison under different n_x on the MNIST-USPS, Fashion, and Multi-COIL-10 datasets.

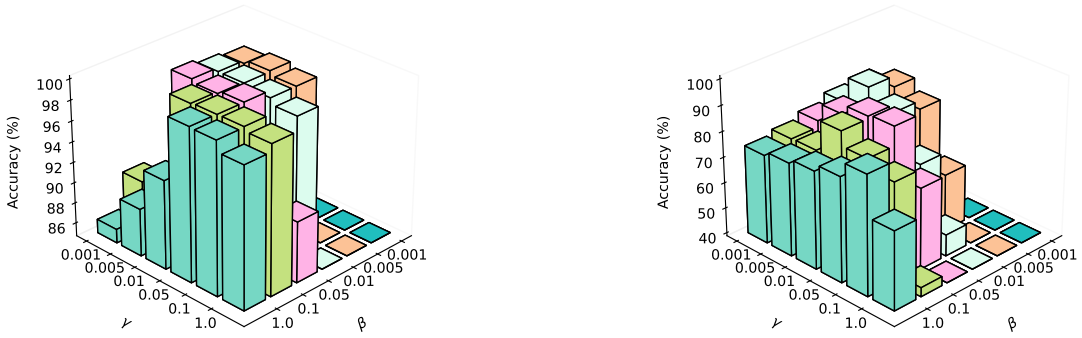
	MNIST-USPS			Fashion			Multi-COIL-10		
Metrics	ACC	NMI	Purity	ACC	NMI	Purity	ACC	NMI	Purity
$n_x = 1$	99.82	99.52	99.82	99.58	98.83	99.58	100.00	100.00	100.00
$n_x = 2$	99.68	99.08	99.68	99.39	98.37	99.39	98.14	96.71	98.14
$n_x = 3$	99.58	98.73	99.58	99.53	98.71	99.53	99.14	98.13	99.14

Network Depth: Inspired by the experimental results presented in Table 4, we carry out ablation experiments on the MNIST-USPS, Fashion, and Multi-COIL-10 datasets as three examples to study the impact of n_x (the number of blocks in GMLM) on the model performance. According to Table 5, we can find that $n_x = 1$ results in the best clustering results. However, the increase in network depth leads to a slight decrease in clustering performance on the three used datasets. It needs to be emphasized that the sizes of the transformation matrices are set to $(49 \times 25, 25 \times 20)$ and $(49 \times 25, 25 \times 20, 20 \times 15)$ under the two settings of $n_x = 2$ and $n_x = 3$, respectively. Therefore, the lose of pivotal structural information in the process of multi-stage deep subspace learning is considered to be the fundamental reason for the degradation of model ability. In summary, these experimental findings not only reaffirm the effectiveness of the designed GMLM for subspace learning again, but also reveal the degradation issue of Grassmannian networks. In the future, we plan to generalize the Euclidean residual learning mechanism to the context of Grassmannian manifolds to mitigate the above problem.

4.5. Parameter analysis

To measure the impact of the trade-off parameters (i.e., α, β, and γ) in Eq (3.13) on the clustering performance of the proposed method, we make experiments on the ORL and MNIST-USPS datasets as two examples. The purpose of introducing these three trade-off parameters to Eq (3.13) is to balance the magnitude of the Grassmannian contrastive learning term L_g, Euclidean contrastive learning term L_c, and regularization term L_a, so as to learning an effective network embedding for clustering. From Figure 2, we have some interesting observations. First, it is not recommended to endow β with a relatively small value. The basic reason is that L_c is not only used to learn discriminative features, but also for the final label prediction. What is more, we can observe that when the value of β is fixed, the clustering accuracy of the proposed method exhibits a trend of first increasing and then decreasing with the change of γ. In addition, it can be found that the value of γ cannot be greater than β. Otherwise, the performance of our method will be significantly affected. The fundamental reason is that a larger γ or a smaller β will cause the gradient information related to L_c to be dramatically weakened in the network optimization process, which is not conducive to learning an effective clustering hypersphere. Besides, we can also observe that the model tends to be less sensitive to β and γ when they vary within the ranges of 0.05~0.005 and 0.01~0.001, respectively. These experimental comparisons support our assertion that the regularization term helps to fine-tune the clustering performance.

In this part, we further investigate the impact of α on the accuracy of the proposed method. We take the MNIST-USPS dataset as an example to conduct experiments, while changing the value of α, and we fix β and γ at their optimal values as shown in Figure 2.



(a) MNIST-USPS (b) ORL

Figure 2. Accuracy comparison under different values of β and γ on the MNIST-USPS and ORL datasets.

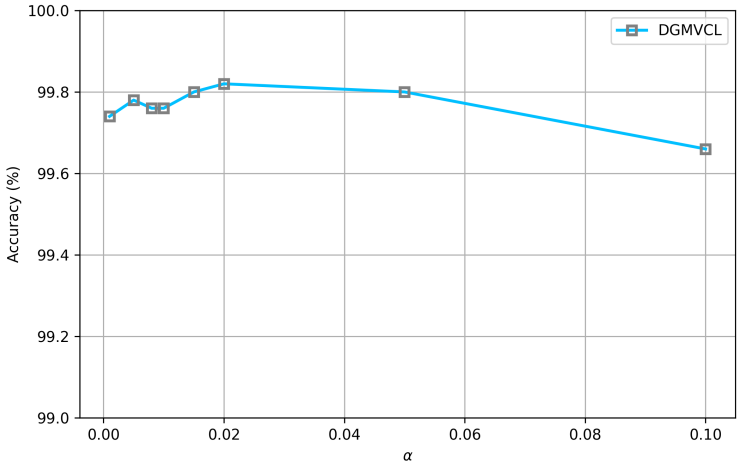


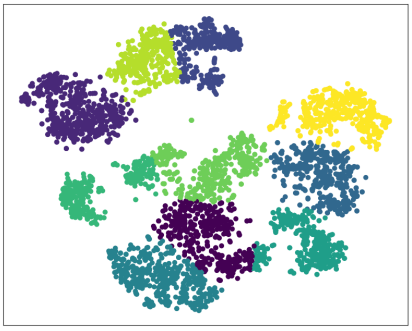
Figure 3. Study the impact of α on the model accuracy on the MNIST-USPS dataset.

It can be seen from Figure 3 that the accuracy of the proposed method generally shows a trend of first increasing and then decreasing, and reaches a peak when α is set to 0.02. Besides, α is not suggested to be endowed with a relatively large value. Otherwise, the gradient information associated with L_c will be weakened in the network optimization process, which is not conducive to the learning of a discriminative network embedding. These experimental results not only further demonstrate the criticality of L_c, but also underscore the effectiveness of α in fine-tuning the discriminability of the learned representations.

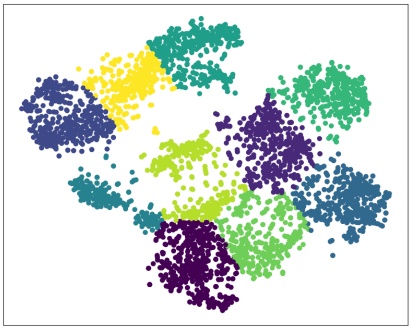
All in all, these experimental observations confirm the complementarity of these three terms in guiding the proposed model to learn more informative features for better clustering. In this paper, our guideline for choosing their values is to ensure that the order of magnitude of the Grassmannian contrastive learning term L_g and the regularization term L_a do not exceed that of the Euclidean contrastive learning term L_c. With this criterion, the model can better integrate the gradient information of L_a and L_g regarding the data distribution with L_c to learn a more reasonable hypersphere for different views. On the MNIST-USPS dataset, the eligible values of α, β, and γ are set to 0.02, 0.05, and 0.005, respectively. We use a similar way to determine that their appropriate values are respectively configured as (0.005, 0.005, 0.005), (0.01, 0.01, 0.01), (0.01, 0.01, 0.005), and (0.1, 0.1, 0.1) on the Fashion, Multi-COIL-10, ORL, and Scene-15 datasets. For a new dataset, the aforementioned principle can help the readers quickly determine the initial value ranges of α, β, and γ.

4.6. Visualization of clustering results

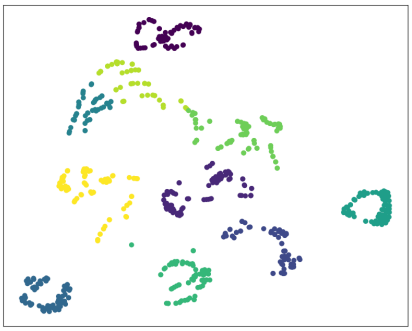
To intuitively test the effectiveness of the proposed method, we select the Fashion and Multi-COIL-10 datasets as two examples to perform 2-D visualization experiments. The experimental results generated by the t-SNE technique [66] are presented in Figure 4, where different colors denote the labels of different clusters. It can be seen that compared to the case where GMLM is not included, the final clustering results, measured by the compactness between similar samples and the diversity between dissimilar samples, are improved by using GMLM on both of the two benchmarking datasets. This further demonstrates that the designed GMLM can help improve the discriminability of cluster assignments.



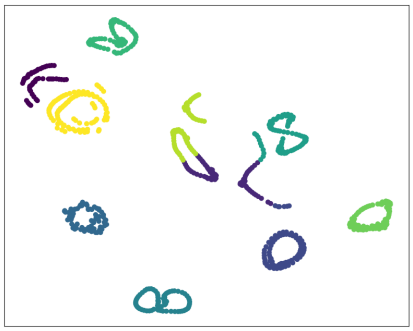
(a) Fashion: "w/o" GMLM



(b) Fashion: with GMLM



(c) Multi-COIL-10: "w/o" GMLM



(d) Multi-COIL-10: with GMLM

Figure 4. Two-dimensional visualization of the clustering results on the Fashion and Multi-COIL-10 datasets with and without using GMLM.

Besides, we further investigate the impact of GMLM on the computational burden of the proposed method. The experimental results are summarized in Table 6, where "w/o" means "without containing". Note that the parameters of the new architecture are kept as the originals.

Table 6. Comparison of training time (s/epoch) on the MNIST-USPS, Fashion, Multi-COIL-10, ORL, and Scene-15 datasets.

Datasets	MNIST-USPS	Fashion	Multi-COIL-10	ORL	Scene-15
DGMVCL-"w/o" GMLM	11.9	34.4	2.6	1.5	14.5
DGMVCL	12.3	36.1	2.7	1.6	16.9

From Table 6, we can see that the integration of GMLM slightly increases the training time of the proposed model across all the five used datasets. According to Section 3, it can be known that the main computational burden of GMLM comes from the QR decomposition used in the ReOrth layer. Nevertheless, as shown in Figure 4, GMLM can enhance the clustering performance of our method, demonstrating its effectiveness.

4.7. *The robustness of the model*

Recent advancements in robust learning, such as projected cross-view learning for unbalanced incomplete multiview clustering [67], have highlighted the importance of handling noisy data and outlier instances. In this section, we evaluate the robustness of the proposed method when subjected to various levels of noise and occlusion, choosing CVCL [55] as a representative competitor. Figure 5 illustrates the accuracy (%) of different methods as a function of the variance of Gaussian noise added to the Scene-15 dataset. It can be seen that as the variance of the Gaussian noise increases, both the proposed method and CVCL experience a decline in accuracy. However, our method consistently outperforms CVCL across all the noise levels. This shows that the suggested model is robust to the Gaussian noise, maintaining good clustering ability even under severe noise levels.

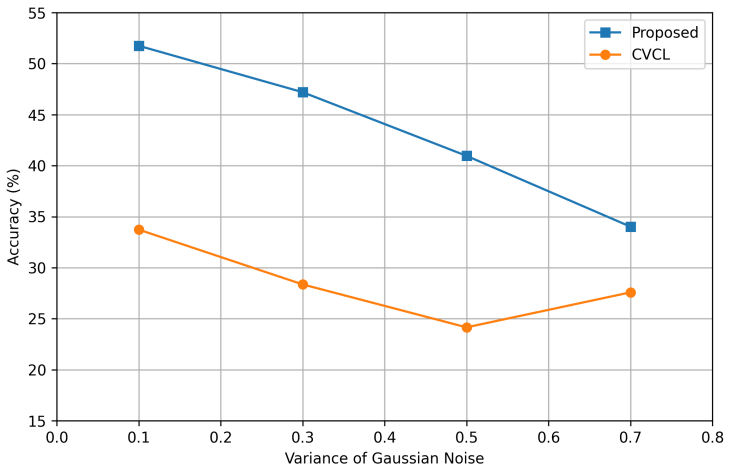


Figure 5. Accuracy of different methods as a function of the variance of Gaussian noise on the Scene-15 dataset.

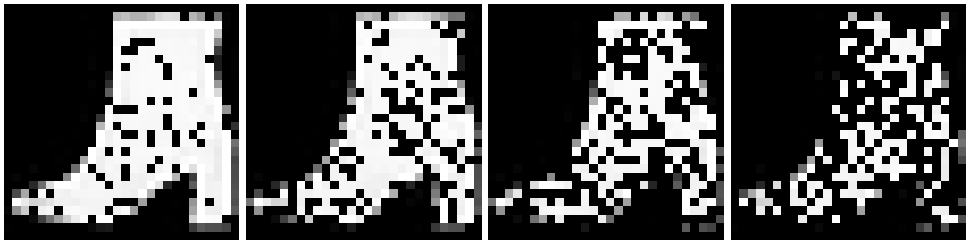


Figure 6. Noisy images with different mask levels of the Fashion dataset.

We further investigate the robustness of the proposed method when handling the data with random pixel masks, selecting the Fashion dataset as an example. Some sample images of the Fashion dataset with 15%, 30%, 45%, and 60% of the pixels masked are shown in Figure 6. According to Table 7, we can see that the performance of the proposed method is superior to that of CVCL under all the corruption levels. These experimental findings again demonstrate that the suggested modifications

over the baseline model are effective to learn more robust and discriminative view-invariant representations. However, as the occlusion rate and the level of Gaussian noise increase, the performance of the proposed DGMVCL shows a noticeable decline. As discussed in Section 3.1.2, each element in the orthogonal basis matrix on the Grassmannian manifold represents the correlation between the original feature dimensions. Although the ability to encode long-range dependencies between different local feature regions enables our model to capture more useful information, it is also susceptible to the influence of local prominent noise. In such a case, the contrastive loss may fail to effectively distinguish between positive and negative samples. This is mainly attributed to the fact that the contrastive learning term treats the decision space as an explicit function of the data distribution. In the future, incorporating techniques like those studied in [67], such as multiview projections or advanced data augmentation, could improve the model's ability to handle these challenges.

Table 7. Accuracy (%) of different methods on the Fashion dataset with random pixel masks.

Methods	15%	30%	45%	60%
CVCL	98.49	97.23	95.06	89.48
DGMVCL	**99.16**	**98.29**	**97.28**	**92.70**

4.8. Discussions

Innovation Analysis: The novelty of our proposed DGMVCL lies not in the mere combination of several existing components but in the thoughtful and innovative way in which these components are integrated and optimized, leading to a lightweight and discriminative geometric learning framework for multiview subspace clustering. Specifically, the suggested DGMVCL introduces several pivotal innovations: i) The Grassmannian neural network, designed for geometric subspace learning, could not be treated as a simple attempt on a new vision application, but rather as an intrinsic method for encoding the underlying submanifold structure of channel features. This is crucial for enabling the model to learn more effective subspace features; ii) The proposed method introduces contrastive learning in both Grassmannian manifolds and Euclidean space. Compared to the baseline model (CVCL [6]) that utilizes the Euclidean-based contrastive loss for network training (in this paper), the additional designed Grassmannian contrastive learning module enables our DGMVCL to characterize and learn the geometrical distribution of the subspace data points more faithfully. Therefore, such a dual-space contrastive learning mechanism is eligible to improve the representational capacity of our model and is capable of extracting view-invariant representations; iii) Extensive evaluations across multiple benchmarking datasets not only demonstrate the superiority of our proposed DGMVCL over the state-of-the-art methods, but also underscores the significance of each individual component and their complementarity.

The Effectiveness of Grassmannian Representation: The Grassmannian manifold is a compact representation of the covariance matrix and encodes the vibrant subspace information, which has shown great success in many applications [11, 27, 28]. Inspired by this, the MMM is designed to capture and parameterize the q-dimensional real vector subspace formed by the features extracted from the FEM. However, the Grassmannian manifold is not a Euclidean space but a Riemannian manifold. We therefore adopt a Grassmannian network to respect the latent Riemannian geometry. Specifically, each network layer can preserve the Riemannian property of the input feature matrices

by normalizing each one into an orthonormal basis matrix. Besides, each manifold-valued weight parameter of the FRMap layer is optimized on a compact Stiefel manifold, not only maintaining its orthogonality but also ensuring better network training. The projector perspective studied in [32] shows that the Grassmannian manifold is an embedded submanifold of the Euclidean space of symmetric matrices, allowing the use of an extrinsic distance, i.e., a projection metric (PM), for measuring the similarity between subspaces over the Grassmannian manifold. In addition, the PM can approximate the true geodesic distance up to a scale factor of $\sqrt{2}$ and is more efficient than the geodesic distance. By leveraging the PM-based contrastive loss, the consistency between cluster assignments across views will be intensified from the Grassmannian perspective while their underlying manifold structure is preserved.

Table 8. Accuracy (%) comparison on the MNIST-USPS, Fashion, Multi-COIL-10, ORL, and Scene-15 datasets.

Datasets	MNIST-USPS	Fashion	Multi-COIL-10	ORL	Scene-15
DGMVCL-"w/o" MMM	10.00	10.00	10.29	2.60	9.14
DGMVCL	99.82	99.52	100.00	92.25	61.29

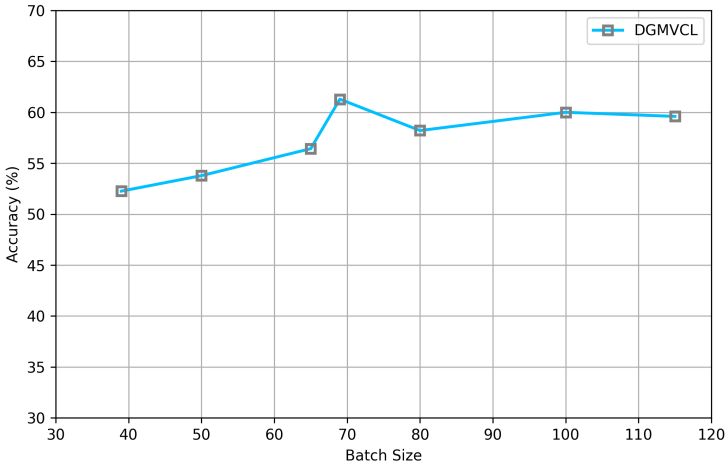


Figure 7. Accuracy of our proposed method under different batch sizes on the Scene-15 dataset.

To intuitively demonstrate the effectiveness of MMM, we conduct a new ablation study to evaluate the clustering ability of the proposed method that does not contain it. Note that the learning rate, batch size, and three trade-off parameters of the new model remain the same as the original, while the size of the input feature matrix of GMLM becomes 49×64. The experimental results on the five used datasets are summarized in Table 8, where "w/o" means "without containing". From Table 8, we can see that removing the designed MMM from DGMVCL leads to a significant decrease in its accuracy across all the five used datasets. This not only confirms the significance of MMM in capturing and parameterizing the underlying submanifold structure, but also reveals the effectiveness of our proposed

model in preserving and leveraging the Riemannian geometry of the data for improved clustering performance.

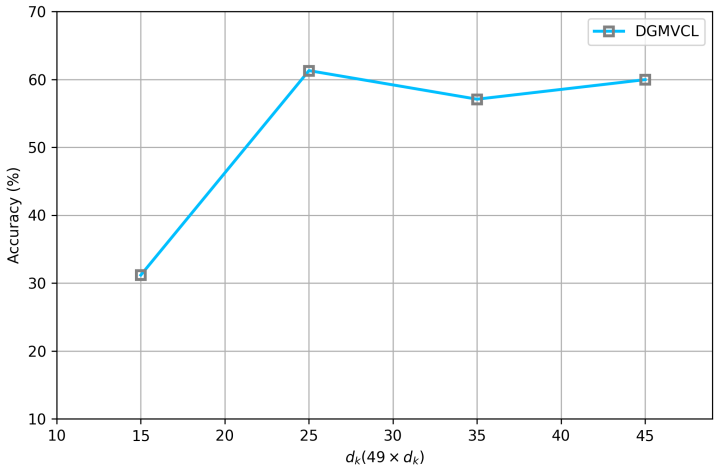


Figure 8. Accuracy of our method under different sizes of the weight matrix in the FRMap layer on the Scene-15 dataset.

Table 9. Accuracy (%) comparison on the Caltech-101 dataset.

Methods	ACC	NMI	Purity
DSIMVC [63]	20.25	31.43	23.68
DCP [64]	27.41	39.58	37.01
CVCL [55]	25.48	37.67	36.63
DGMVCL	29.03	45.81	40.02

Selection of Network Parameters: The selection of network parameters is based on experiments and analysis to ensure optimal outcomes. Specifically, the trade-off parameters α, β, and γ play a critical role in balancing the contributions of different loss functions in the overall objective function. The magnitude of the pivotal loss functions should be slightly higher. Based on this guideline, we can roughly determine their initial value, signified as the anchor point. Then, a candidate set can be formed around the selected anchor point. After that, we can conduct experiments to determine their optimal values. Additionally, the learning rate and batch size are crucial for the convergence and effectiveness of the proposed model. A too-high learning rate might cause the model to diverge, while a too-low one would slow down the training process [68]. Since CVCL [55] is our base model, we treat its learning rate as the initial value and adjust around it to find the suitable one. The batch size is configured to balance the memory usage and training efficiency. For the Scene-15 dataset, a batch size of 69 was chosen because this dataset contains 4485 samples, and 69 divides this number evenly, ensuring efficient utilization of data in each batch. Additionally, as shown in Figure 7, this batch size can yield good performance. However, in practice, the batch size is also related to the computing device. On the MNIST-USPS, Multi-COIL-10, Fashion, ORL, and Scene-15 datasets, the learning rate and batch size are specifically configured as (0.0002, 50), (0.0001, 50), (0.0005, 100), (0.0001,

50), and (0.001, 69), respectively. Furthermore, the experimental results presented in Figure 8 suggest that it is appropriate to configure the size of the transformation matrix in the FRMap layer as 49×25. When d_k is assigned to a small value, some useful geometric information will be lost during feature transformation mapping. In contrast, a relatively larger d_k results in more redundant information being incorporated into the generated subspace features. Both of these two cases have a negative impact on the model performance.

In short, the choice of network parameters are supported by both theoretical considerations and empirical evidence, and they contribute to the overall performance of the proposed model.

Other Datasets: In this part, the Caltech-101 dataset [39] has been applied to further evaluate the effectiveness of the proposed model. This dataset is a challenging benchmark for object detection, which consists of 101 different object categories as well as one background category, totaling approximately 9146 images.

The experimental results achieved by different comparative models on the Caltech-101 dataset are listed in Table 9. Note that the learning rate, batch size, and the size of the weight matrix in the FRMap layer of the proposed DGMVCL are configured as 0.005, 50, and 49×25, respectively. According to Table 9, we can see that the clustering accuracy of our proposed DGMVCL are 8.87%, 1.62%, and 3.55% higher than that of DSIMVC, DCP, and CVCL, respectively. Additionally, under the other two validation metrics, i.e., NMI and purity, our method is still the best performer. This demonstrates that the suggested Grassmannian manifold-valued deep contrastive learning mechanism can learn compact and discriminative geometric features for MVC, even in complicated data scenarios.

5. Conclusions

In this paper, a novel framework is suggested to learn view-invariant representations for multiview clustering (MVC), called DGMVCL. Considering the submanifold structure of channel features, a Grassmannian neural network is constructed for the sake of characterizing and learning the subspace data more faithfully and effectively. Besides, the contrastive learning mechanism built upon the Grassmannian manifold and Euclidean space enables more discriminative cluster assignments. Extensive experiments and ablation studies conducted on five MVC datasets not only demonstrate the superiority of our proposed method over the state-of-the-art methods, but also confirm the usefulness of each designed component.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (62306127, 62020106012, 62332008, U1836218), the Natural Science Foundation of Jiangsu Province (BK20231040), the Fundamental Research Funds for the Central Universities (JUSRP124015), the Key Project of Wuxi Municipal Health Commission (Z202318), and the National Key R&D Program of China (2023YFF1105102, 2023YFF1105105).

Conflict of interest

The authors declare no conflict of interest.

References

1. M. C. Tsakiris, R. Vidal, Algebraic clustering of affine subspaces, *IEEE Trans. Pattern Anal. Mach. Intell.*, **40** (2017), 482–489. https://doi.org/10.1109/TPAMI.2017.2678477
2. C. You, C. G. Li, D. P. Robinson, R. Vidal, Is an affine constraint needed for affine subspace clustering?, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, (2019), 9915–9924.
3. P. Ji, M. Salzmann, H. Li, Shape interaction matrix revisited and robustified: Efficient subspace clustering with corrupted and incomplete data, in *Proceedings of the IEEE International Conference on computer Vision*, (2015), 4687–4695.
4. J. Yang, J. Liang, K. Wang, P. L. Rosin, M. H. Yang, Subspace clustering via good neighbors, *IEEE Trans. Pattern Anal. Mach. Intell.*, **42** (2019), 1537–1544. https://doi.org/10.1109/TPAMI.2019.2913863
5. A. Gruber, Y. Weiss, Multibody factorization with uncertainty and missing data using the EM algorithm, in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (2004), 1–1. https://doi.org/10.1109/CVPR.2004.1315101
6. S. R. Rao, R. Tron, R. Vidal, Y. Ma, Motion segmentation via robust subspace separation in the presence of outlying, incomplete, or corrupted trajectories, in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, (2008), 1–8. https://doi.org/10.1109/CVPR.2008.4587437
7. E. Elhamifar, R. Vidal, Sparse subspace clustering: Algorithm, theory, and applications, *IEEE Trans. Pattern Anal. Mach. Intell.*, **35** (2013), 2765–2781. https://doi.org/10.1109/TPAMI.2013.57
8. Z. Kang, G. Shi, S. Huang, W. Chen, X. Pu, J. T. Zhou, et al., Multi-graph fusion for multi-view spectral clustering, *Knowl.-Based Syst.*, **189** (2020), 105102. https://doi.org/10.1016/j.knosys.2019.105102
9. G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, Y. Ma, Robust recovery of subspace structures by low-rank representation, *IEEE Trans. Pattern Anal. Mach. Intell.*, **35** (2012), 171–184. https://doi.org/10.1109/TPAMI.2012.88
10. J. Shi, J. Malik, Normalized cuts and image segmentation, *IEEE Trans. Pattern Anal. Mach. Intell.*, **22** (2000), 888–905. https://doi.org/10.1109/34.868688
11. J. Guo, Y. Sun, J. Gao, Y. Hu, B. Yin, Low rank representation on product grassmann manifolds for multi-view subspace clustering, in *2020 25th International Conference on Pattern Recognition (ICPR)*, (2021), 907–914. https://doi.org/10.1109/ICPR48806.2021.9412242
12. W. B. Hu, X. J. Wu, Multi-geometric sparse subspace clustering, *Neural Process. Lett.*, **52** (2020), 849–867. https://doi.org/10.1007/s11063-020-10274-z
13. D. Wei, X. Shen, Q. Sun, X. Gao, Discrete metric learning for fast image set classification, *IEEE Trans. Image Process.*, **31** (2022), 6471–6486. https://doi.org/10.1109/TIP.2022.3212284

14. G. Cheng, C. Yang, X. Yao, L. Guo, J. Han, When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative CNNs, *IEEE Trans. Geosci. Remote Sens.*, **56** (2018), 2811–2821. https://doi.org/10.1109/TGRS.2017.2783902

15. K. Song, J. Han, G. Cheng, J. Lu, F. Nie, Adaptive neighborhood metric learning, *IEEE Trans. Pattern Anal. Mach. Intell.*, **44** (2021), 4591–4604.

16. P. Ji, T. Zhang, H. Li, M. Salzmann, I. Reid, Deep subspace clustering networks, *Adv. Neural Inf. Process. Syst.*, **30** (2017).

17. H. Wang, Q. Wang, Q. Miao, X. Ma, Joint learning of data recovering and graph contrastive denoising for incomplete multi-view clustering, *Inf. Fusion*, **104** (2024), 102155. https://doi.org/10.1016/j.inffus.2023.102155

18. W. Wu, X. Ma, Q. Wang, M. Gong, Q. Gao, Learning deep representation and discriminative features for clustering of multi-layer networks, *Neural Networks*, **170** (2024), 405–416. https://doi.org/10.1016/j.neunet.2023.11.053

19. J. Xu, Y. Ren, G. Li, L. Pan, C. Zhu, Z. Xu, Deep embedded multi-view clustering with collaborative training, *Inf. Sci.*, **573** (2021). 279–290. https://doi.org/10.1016/j.ins.2020.12.073

20. H. Wang, W. Zhang, X. Ma, Contrastive and adversarial regularized multi-level representation learning for incomplete multi-view clustering, *Neural Networks*, **172** (2024), 106102. https://doi.org/10.1016/j.neunet.2024.106102

21. Y. Yang, X. Ma, Graph contrastive learning for clustering of multi-layer networks, *IEEE Trans. Big Data*, **2023** (2023). https://doi.org/10.1109/TBDATA.2023.3343349

22. W. Guo, H. Che, M. F. Leung, Z. Yan, Adaptive multi-view subspace learning based on distributed optimization, *Int. Things*, **26** (2024), 101203. https://doi.org/10.1016/j.iot.2024.101203

23. Z. Li, Q. Wang, Z. Tao, Q. Gao, Z. Yang, Deep adversarial multi-view clustering network, in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, **2** (2019), 4.

24. H. Ma, W. Wu, A deep clustering framework integrating pairwise constraints and a VMF mixture model, *Electron. Res. Arch.*, **32** (2024), 3952–3972. https://dx.doi.org/10.3934/era.2024177

25. Z. Chen, T. Xu, X. J. Wu, R. Wang, Z. Huang, J. Kittler, Riemannian local mechanism for spd neural networks, in *Proceedings of the AAAI Conference on Artificial Intelligence*, **37** (2023), 7104–7112. https://doi.org/10.1609/aaai.v37i6.25867

26. R. Wang, X. J. Wu, Z. Chen, C. Hu, J Kittler Spd manifold deep metric learning for image set classification, *IEEE Trans. Neural Networks Learn. Syst.*, **35** (2024), 8924–8938. https://doi.org/10.1109/TNNLS.2022.3216811

27. T. Liu, Z. Shi, Y. Liu, Visual clustering based on kernel sparse representation on grassmann manifolds, in *2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, (2017), 920–925. https://doi.org/10.1109/CYBER.2017.8446507

28. R. Wang, X. J. Wu, Z. Liu, J. Kittler, Geometry-aware graph embedding projection metric learning for image set classification, *IEEE Trans. Cognitive Dev. Syst.*, **14** (2022), 957–970. https://doi.org/10.1109/TCDS.2021.3086814

29. R. Wang, X. J. Wu, J. Kittler, Graph embedding multi-kernel metric learning for image set classification with Grassmannian manifold-valued features, *IEEE Trans. Multimedia*, **23** (2021), 228–242. https://doi.org/10.1109/TMM.2020.2981189

30. R. Wang, X. J. Wu, T. Xu, C. Hu, J. Kittler, U-SPDNet: An SPD manifold learning-based neural network for visual classification, *Neural networks*, **161** (2023), 382–396. https://doi.org/10.1016/j.neunet.2022.11.030

31. K. X. Chen, X. J. Wu, R. Wang, J. Kittler, Riemannian kernel based Nyström method for approximate infinite-dimensional covariance descriptors with application to image set classification, in *2018 24th International Conference on Pattern Recognition (ICPR)*, (2018), 651–656. https://doi.org/10.1109/ICPR.2018.8545822

32. T. Bendokat, R. Zimmermann, P. A. Absil, A grassmann manifold handbook: Basic geometry and computational aspects, *Adv. Comput. Math.*, **50** (2024), 1–51. https://doi.org/10.1007/s10444-023-10090-8

33. D. Wei, X. Shen, Q. Sun, X. Gao, Z. Ren, Sparse representation classifier guided Grassmann reconstruction metric learning with applications to image set analysis, *IEEE Trans. Multimedia*, **25** (2022), 4307–4322. https://doi.org/10.1109/TMM.2022.3173535

34. B. Wang, Y. Hu, J. Gao, Y. Sun, B. Yin, Low rank representation on Grassmann manifolds, in *Computer Vision–ACCV 2014: 12th Asian Conference on Computer Vision*, (2015), 81–96. https://doi.org/10.1007/978-3-319-16865-4_6

35. X. Piao, Y. Hu, J. Gao, Y. Sun, B. Yin, Double nuclear norm based low rank representation on Grassmann manifolds for clustering, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2019), 12075–12084.

36. C. Zhang, Q. Hu, H. Fu, P. Zhu, X. Cao, Latent multi-view subspace clustering, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (2017), 4279–4287.

37. R. Li, C. Zhang, Q. Hu, P. Zhu, Z. Wang, Flexible multi-view representation learning for subspace clustering, in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, (2019), 2916–2922.

38. R. Zhou, Y. D. Shen, End-to-end adversarial-attention network for multi-modal clustering, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2020), 14619–14628.

39. Z. Kang, Z. Lin, X. Zhu, W. Xu, Structured graph learning for scalable subspace clustering: From single view to multiview, *IEEE Trans. Cybern.*, **52** (2021), 8976–8986. https://doi.org/10.1109/TCYB.2021.3061660

40. E. Pan, Z. Kang, High-order multi-view clustering for generic data, *Inf. Fusion*, **100** (2023), 101947. https://doi.org/10.1016/j.inffus.2023.101947

41. J. Chen, S. Yang, X. Peng, D. Peng, Z. Wang, Augmented sparse representation for incomplete multiview clustering, *IEEE Trans. Neural Networks Learn. Syst.*, **35** (2022), 4058–4071. https://doi.org/10.1109/TNNLS.2022.3201699

42. T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A simple framework for contrastive learning of visual representations, in *International Conference on Machine Learning*, (2020), 1597–1607.

43. Y. Tian, C. Sun, B. Poole, D. Krishnan, C. Schmid, P. Isola, What makes for good views for contrastive learning?, *Adv. Neural Inf. Process. Syst.*, **33** (2020), 6827–6839.
44. T. Wang, P. Isola, Understanding contrastive representation learning through alignment and uniformity on the hypersphere, in *International Conference on Machine Learning*, (2020), 9929–9939.
45. M. Harandi, C. Sanderson, C. Shen, B. C. Lovell, Dictionary learning and sparse coding on Grassmann manifolds: An extrinsic solution, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, (2013), 3120–3127.
46. Z. Huang, J. Wu, L. Van Gool, Building deep networks on Grassmann manifolds, in *Proceedings of the AAAI Conference on Artificial Intelligence*, (2018), 1137–1145. https://doi.org/10.1609/aaai.v32i1.11725
47. A. Edelman, T. A. Arias, S. T. Smith, The geometry of algorithms with orthogonality constraints, *SIAM J. Matrix Anal. Appl.*, (1998), 303–353. https://doi.org/10.1137/S0895479895290954
48. P. A. Absil, R. Mahony, R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*, Princeton University Press, 2009.
49. J. Hamm, D. D. Lee, Grassmann discriminant analysis: A unifying view on subspace-based learning, in *Proceedings of the 25th International Conference on Machine Learning*, (2008), 376–383. https://doi.org/10.1145/1390156.1390204
50. J. Hamm, D. Lee, Extended Grassmann kernels for subspace-based learning, *Adv. Neural Inf. Process. Syst.*, (2008), 21.
51. M. T. Harandi, C. Sanderson, S. Shirazi, B. C. Lovell, Graph embedding discriminant analysis on Grassmannian manifolds for improved image set matching, in *CVPR 2011*, (2011), 2705–2712. https://doi.org/10.1109/CVPR.2011.5995564
52. Z. Huang, S. Shan, R. Wang, H. Zhang, S. Lao, A. Kuerban, et al., A benchmark and comparative study of video-based face recognition on cox face database, *IEEE Trans. Image Process.*, **24** (2015), 5967–5981. https://doi.org/10.1109/TIP.2015.2493448
53. C. Cui, Y. Ren, J. Pu, X. Pu, L. He, Deep multi-view subspace clustering with anchor graph, preprint, arXiv:2305.06939. https://doi.org/10.48550/arXiv.2305.06939
54. P. Xia, L. Zhang, F. Li, Learning similarity with cosine similarity ensemble, *Inf. Sci.*, **307** (2015), 39–52. https://doi.org/10.1016/j.ins.2015.02.024
55. J. Chen, H. Mao, W. L. Woo, X. Peng, Deep multiview clustering by contrasting cluster assignments, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, (2023), 16752–16761.
56. A. Asuncion, D. Newman, *UCI Machine Learning Repository*, 2007. Available from: https://ergodicity.net/2013/07/
57. J. Xu, Y. Ren, H. Tang, X. Pu, X. Zhu, M. Zeng, et al., Multi-VAE: Learning disentangled view-common and view-peculiar visual representations for multi-view clustering, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, (2021), 9234–9243.
58. H. Xiao, K. Rasul, R. Vollgraf, Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms, preprint, arXiv:1708.07747. https://doi.org/10.48550/arXiv.1708.07747

59. M. D. Addlesee, A. Jones, F. Livesey, F. Samaria, The ORL active floor [sensor system], *IEEE Pers. Commun.*, **4** (1997), 35–41. https://doi.org/10.1109/98.626980

60. F. F. Li, P. Perona, A bayesian hierarchical model for learning natural scene categories, in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, **2** (2005), 524–531. https://doi.org/10.1109/CVPR.2005.16

61. U. Von Luxburg, A tutorial on spectral clustering, *Stat. Comput.*, **17** (2007), 395–416. https://doi.org/10.1007/s11222-007-9033-z

62. F. Nie, X. Wang, H. Huang, Clustering and projected clustering with adaptive neighbors, in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (2014), 977–986. https://doi.org/10.1145/2623330.2623726

63. H. Tang, Y. Liu, Deep safe incomplete multi-view clustering: Theorem and algorithm, in *International Conference on Machine Learning*, (2022), 21090–21110.

64. Y. Lin, Y. Gou, X. Liu, J. Bai, J. Lv, X. Peng, Dual contrastive prediction for incomplete multi-view representation learning, *IEEE Trans. Pattern Anal. Mach. Intell.*, **45** (2022), 4447–4461. https://doi.org/10.1109/TPAMI.2022.3197238

65. J. Xu, H. Tang, Y. Ren, L. Peng, X. Zhu, L. He, Multi-level feature learning for contrastive multi-view clustering, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2022), 16051–16060.

66. L. Van der Maaten, G. Hinton, Visualizing data using t-SNE., *J. Mach. Learn. Res.*, **9** (2008).

67. Y. Cai, H. Che, B. Pan, M. F. Leung, C. Liu, S. Wen, Projected cross-view learning for unbalanced incomplete multi-view clustering, *Inf. Fusion*, **105** (2024), 102245. https://doi.org/10.1016/j.inffus.2024.102245

68. K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, inn *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2016), 770–778.

AIMS Press