Deep Inductive and Scalable Subspace Clustering via Nonlocal Contrastive Self-Distillation

Wenjie Zhu, *Member, IEEE,* Bo Peng, and Wei Qi Yan, *Senior Member, IEEE*

***Abstract*—Deep subspace clustering has demonstrated remarkable results by leveraging the nonlinear subspace assumption. However, it often encounters challenges in terms of computational cost and memory footprint in dealing with large-scale data due to its traditional single-batch training strategy. To address this issue, this paper proposes a deep subspace clustering framework that is regularized by nonlocal contrastive self-distillation, enabling a Deep Inductive and Scalable Subspace Clustering (DISSC) algorithm. In particular, our framework incorporates two subspace learning modules, namely subspace learning based on self-expression model and inductive subspace clustering. These modules generate affinities from different perspectives by extracting intermediate features from two augmentations of the input data using a weight-sharing neural network. By integrating the concept of self-distillation, our framework effectively exploits the clustering-friendly knowledge contained in these two affinities through a novel nonlocal contrastive prediction task, employing an empirical yet effective threshold. This allows the framework to facilitate complementary knowledge mining and scalability without compromising clustering performance. With an alternate branch that bypasses the self-expression computation, our framework can infer subspace membership of the out-of-sample data through the predicted soft labels, eliminating the need for ad-hoc postprocessing. In addition, the self-expression matrix computed using mini-batch data benefits from the distilled knowledge obtained from the inductive subspace clustering module, enabling our framework to scale to data of arbitrary size. Experiments conducted on large-scale MNIST, Fashion-MINST, STL-10, CIFAR-10 and Stanford Online Products datasets validate the superiority of the proposed DISSC algorithm over state-of-the-art subspace clustering methods.**

***Index Terms*—Deep subspace clustering, large-scale data, scalability, inductive clustering, knowledge self-distillation.**

I. Introduction

With the expeditious development of 5G technology, an increasing number of smart devices have been connected to the Internet, constantly injecting a vast amount of data. This surge in data has led to a higher computational demand for real-time clustering analysis. Subspace clustering, as one of the most prominent clustering algorithms, aims to partition a set of data points approximately drawn from a union of low-dimensional linear subspaces into disjoint groups, ensuring that all data points within a given group lie in the same subspace. Owing to its robust performance in handling high-dimensional data with outliers, subspace clustering methods have found successful applications in data mining [1], [2], [3], [4] and computer vision [5], [6], [7], [8].

Over the years, numerous subspace clustering methods have been developed based on the self-expression property. Typically, these methods involve a two-step process. Firstly, an affinity matrix is constructed between data points based on the self-expression model, followed by the application of spectral clustering to this matrix. Previous efforts in subspace clustering have utilized the self-expression property with sparse [1], [9], [10], low-rank [2], [3], [11], [12], collaborative [13], [14] or entropy [4], [15], [16] regularization to learn pairwise affinities while assuming a linear subspace. However, real-world data often deviates from linear models. Consequently, researchers have explored kernel-based subspace clustering methods [17], [18], [19], [20] to establish a self-expression model that accommodates nonlinear assumptions. Although the kernel trick can address the issue of nonlinear subspaces, defining an appropriate kernel function that captures the underlying nonlinear structure of the data is often a challenging task. Different datasets may require different kernel functions, and choosing the right kernel that represents the data can be subjective and problem-specific. Additionally, there is no guarantee that the selected kernel will effectively model the nonlinear subspace structure inherent in the data.

In recent years, there has been a remarkable upsurge of interest in deep subspace clustering approaches. These approaches harness the immense potential of deep neural networks as powerful tools for nonlinear mapping. By exploiting the capabilities of deep neural networks, researchers aim to uncover latent feature spaces that are highly conducive to effective subspace clustering [21], [22], [23], [24]. One prominent approach that has gained significant attention involves training an autoencoder with a meticulously crafted self-expression layer. By incorporating a self-expression layer into the training process, the autoencoder becomes capable of extracting meaningful features from the input data. These extracted features are subsequently utilized to construct an affinity matrix, which encodes the relationships between data points. The integration of the self-expression layer within the autoencoder framework has proven to be particularly effective

Manuscript received **, 2023; revised **, 2023.

This research was partially supported by Zhejiang Provincial Natural Science Foundation of China under Grant No.LY24F030005, National Key Research and Development Program of China under Grant No.2021YFC3340402, and the Fundamental Research Funds for the Provincial Universities of Zhejiang under Grant No.2022YW40.

Wenjie Zhu is with Zhejiang-New Zealand Joint Vision-Based Intelligent Metrology Laboratory, College of Information Engineering, China Jiliang University, Hangzhou 310018, China. He is a visiting scholar at Auckland University of Technology, Auckland 1010, New Zealand (e-mail: zhu.wenjie@aut.ac.nz).

Bo Peng was with The University of Queensland, St. Lucia 4072, Australia.

Wei Qi Yan is with Zhejiang-New Zealand Joint Vision-Based Intelligent Metrology Laboratory, Department of Computer Science, Auckland University of Technology, Auckland 1010, New Zealand (e-mail: wyan@aut.ac.nz).

in achieving accurate subspace clustering results. The two most representative methods are deeP subspAce clusteRing with sparsiTY prior (PARTY) [21] and Deep Subspace Clustering Networks (DSC-Net) [23]. PARTY incorporates prior sparsity information into the hidden representation learning process, which is achieved by minimizing the reconstruction error of the input data. By considering sparsity as a prior, PARTY aims to discover meaningful features that effectively capture the underlying subspace structure. In contrast, DSC-Net employs a convolutional autoencoder to simultaneously learn the intermediate features and the expression matrix. By jointly optimizing the representation learning process and the relationship between data points in the latent space, DSC-Net seeks to enhance the clustering accuracy within the subspaces. Both PARTY and DSC-Net contribute to the advancement of subspace clustering methods by leveraging deep learning techniques. Subsequently, There is a growing emphasis among researchers [25], [26] on improving the data representations and/or affinities based on the DSC-Net framework.

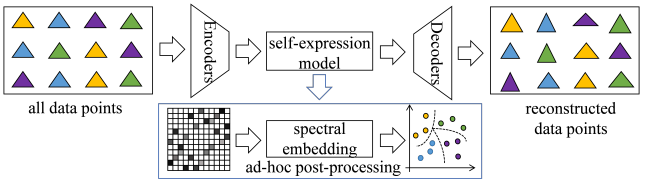
Although the autoencoder-based DSC-Net and its subsequent approaches have made significant contributions to the advancement of deep subspace clustering, important considerations remain as follows:

1) *Autoencoder-based backbones*. The current deep subspace clustering methods primarily focus on achieving high pixel-level reconstruction accuracy with the autoencoder-based backbones. However, this approach may not effectively capture the underlying structural representations required for the subspace clustering task. It may fail to reflect the intrinsic structure of datasets, which is crucial for capturing the subspace-preserving property.

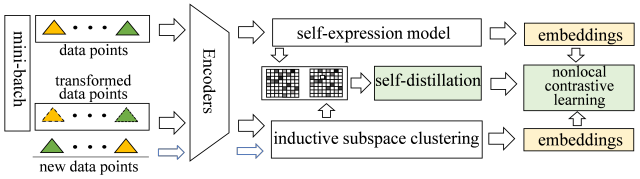
2) *Single-batch training strategy*. The existing autoencoder-based deep subspace clustering approaches employ spectral clustering on the induced affinity matrix for data segmentation. Unfortunately, this step requires an excessively large batch size to construct the affinity matrix. This scalability issue becomes a challenge when dealing with large-scale data, as it consumes significant memory and computation resources (with a complexity of $\mathcal{O}(n^2)$ for memory and $\mathcal{O}(kn^2)$ for matrix decomposition), where n denotes the number of data points and k represents the number of clusters.

3) *Training from scratch for unseen data*. In deep subspace clustering, the self-expression layer generates a quadratic-size coefficient matrix, from which cluster labels are extracted using spectral clustering in a post-processing step. Consequently, the transductive clustering approach is infeasible for training on unseen data. Therefore, when new data is involved in the clustering analysis, the deep subspace clustering procedure needs to be repeated to incorporate this new information.

In light of the aforementioned discussion, the widely used encoder-decoder framework may not exhibit significant advantages in achieving scalable and inductive deep subspace clustering for large-scale datasets. In light of this, this paper proposes a novel framework called Deep Inductive and Scalable Subspace Clustering (DISSC) with nonlocal contrastive self-distillation. Unlike the DSC-Net and its variants that employ autoencoders to obtain meaningful representations of the whole data in Fig. 1, our DISSC framework introduces two modules, namely self-expression subspace learning and inductive subspace clustering, implemented through separate branches.



(a) Autoencoder-based DSC networks



(b) Proposed DISSC framework

Fig. 1. Comparison between the proposed DISSC framework and autoencoder-based deep subspace clustering networks. Our DISSC framework differs from autoencoder-based DSC networks, which utilize a single-batch training strategy involving all data points and a transductive clustering mechanism that minimizes the pixel-level reconstruction error. Instead, our method considers mini-batch data points with two different transformations. The proposed DISSC decouples two functions, namely self-expression subspace learning and inductive subspace clustering, and constrain them using a novel nonlocal contrastive self-distillation loss. The inference for new data points can be conducted as shown by the blue arrows.

From an inductive clustering perspective, it is evident from Fig. 1 that the transductive clustering paradigm and the computation of the self-expression matrix using all data points hinder performance in inductive clustering. A viable solution to this issue is decoupling the subspace learning function, which involves creating a bypass for inductive clustering. This approach allows the data stream to flow through the inductive clustering branch without passing through the self-expression layer during inference. Consequently, the ad-hoc post-processing step following network training can be eliminated. In terms of scalability, a preferred mini-batch training strategy can reduce memory consumption and computational storage. However, the subspace spanned by the limited number of data points may not adequately capture the global structure of the dataset. To address this limitation, drawing inspiration from knowledge distillation, a nonlocal contrastive loss with self-distillation is additionally proposed to enhance the subspaces generated by the two modules. Detailed information about the two modules can be found in Section III. To sum up, the contributions of this work can be summarized as follows:

1) An inductive and scalable deep subspace clustering framework is proposed with contrastive self-distillation by designing the subspace learning and inductive subspace clustering modules. By decoupling the computation of the self-expression matrix and the inductive clustering process, our framework provides valuable insights into deep subspace clustering in large-scale clustering scenarios.
2) The principle of contrastive self-distillation is leveraged to enhance meaningful subspace information with mini-

batch data and reliable subspace membership prediction without ground-truth labels. This highlights the natural complementarity of self-distillation in inductive and scalable subspace clustering.

3) Extensive experimental results on several large-scale benchmark datasets, i.e., MNIST, Fashion-MNIST, and Stanford Online Products, demonstrate the superiority of our framework over the state-of-the-art deep subspace clustering methods. Moreover, our framework exhibits great potential in inductive clustering for the out-of-sample data.

The remainder of this paper is organized as follows: Section II provides a brief review of the related work, including deep subspace clustering and knowledge self-distillation. Section III elaborates on the proposed method, presenting its framework and architecture in detail. Section IV outlines the experimental setup and presents the results and analysis. Finally, Section V concludes this work, summarizing the contributions and discussing avenues for future research.

II. Related Work

Clustering is an effective technique used to group similar data points together while separating dissimilar ones [27], with applications spanning computer vision [28], [29], pattern recognition [30], and beyond. Subspace clustering algorithm based on self-expression model was initially proposed in [9]. This algorithm assumes that each data point within a union of subspaces can be expressed as a linear combination of other data points belonging to the same subspaces. With the self-expression coefficient matrix derived from the data, it becomes possible to construct an affinity matrix that captures the relationships between every pair of data points. Spectral clustering can then be applied to this affinity matrix to perform clustering on the data. Based on the utilization of deep learning techniques, existing works in this field can be classified into two categories: traditional subspace clustering and deep subspace clustering. To facilitate clarity and ease of understanding, this paper adopts a convention where matrices and vectors are denoted by bold uppercase/lowercase letters, such as $\boldsymbol{X}$ and $\boldsymbol{x}$, respectively. Moreover, detailed explanations of all notations used in this paper can be found in Table I.

A. Traditional subspace clustering

The traditional subspace clustering methods, which utilize sparse, low-rank, or a combination of both regularization techniques, have gained significant popularity in terms of simplicity, theoretical foundations, and empirical achievements. Mathematically, the optimization problem can be formulated as follows:

$$\min_{\boldsymbol{C}} \|\boldsymbol{X} - \boldsymbol{X}\boldsymbol{C}\|_{\mathrm{F}}^2 + \mathtt{reg}\left(\boldsymbol{C}\right), \tag{1}$$

where $\boldsymbol{X} \in \mathbb{R}^{d\times n}$ is the collection of n data points in the d-dimensional feature space, the regularization function $\mathtt{reg}(\cdot)$ is with various assumptions on the self-expression matrix $\boldsymbol{C} \in \mathbb{R}^{n\times n}$, such as the ℓ_1 norm, ℓ_2 norm, and nuclear norm. Subsequently, the affinity matrix $\boldsymbol{A}$ can be computed as $\boldsymbol{A} = \left(|\,\boldsymbol{C}\,| + |\,\boldsymbol{C}\,|^{\top}\right)/2$, facilitating spectral clustering. Nevertheless, traditional subspace clustering methods face two crucial challenges that have a substantial impact on both theoretical analysis and practical implementation: the restrictive assumption of linear subspaces and the computational complexity associated with large-scale convex optimization. In light of these two aspects, several researchers have conducted extensive investigations and made significant contributions to addressing these issues.

TABLE I
Notations in this paper

Notation	Description
$\mathcal{X} \in \mathbb{R}^{m\times n\times c}$	A 3-order Tensor $\mathcal{X}$
$\mathcal{X}^j \in \mathbb{R}^{m\times n}$	The j-th element of the tensor $\mathcal{X}$
$\boldsymbol{X} \in \mathbb{R}^{m\times n}$	An m by n matrix
$\boldsymbol{X}^j \in \mathbb{R}^{m\times 1}$	The j-th column of matrix $\boldsymbol{X}$
$\boldsymbol{X}\,(i,j)$	The i-th row and the j-th column of the matrix $\boldsymbol{X}$
$\|\cdot\|$	ℓ_2 norm of a vector
$\|\cdot\|_{\mathrm{F}}$	Frobenious norm of a matrix
$(\cdot)^{\top}$	Transposed matrix
$\mathcal{P}\,(i)$	Positive set of the data point indexed by i
$\mathcal{N}\,(i)$	Negative set of the data point indexed by i

Traditional subspace clustering algorithms assume that the data is generated from a collection of linear subspaces, which limits their capability to effectively handle nonlinear subspace clustering problems. Although the kernel trick [31] can be employed to address such nonlinear scenarios, selecting an appropriate kernel becomes time-consuming in real-world applications. Consequently, to compensate for the limitations of using a single kernel, various subspace clustering algorithms have related to the concept of multiple kernels learning [17]. Furthermore, the computational time and memory demands are increased by the data self-expression model during the solution of large-scale convex optimization problems. To mitigate these challenges, researchers have explored scalable subspace clustering methods [9], [32], [33], [34] in recent years.

B. Deep subspace clustering

Deep subspace clustering methods leverage neural networks as a combination of nonlinear mapping functions to handle the nonlinear assumptions inherent in datasets. The corresponding optimization problem can be expressed as:

$$\min_{\boldsymbol{C}} \|f(\boldsymbol{X}) - f(\boldsymbol{X})\boldsymbol{C}\|_{\mathrm{F}}^2 + \mathtt{reg}\left(\boldsymbol{C}\right) + \|\boldsymbol{X} - \hat{\boldsymbol{X}}\|_{\mathrm{F}}^2. \tag{2}$$

In Eqn. (2), the reconstructed data $\hat{\boldsymbol{X}}$ can be computed as $\hat{\boldsymbol{X}} = g(f(\boldsymbol{X}))$, where $f(\cdot)$ and $g(\cdot)$ represent the mapping function implemented by encoders and decoders respectively. To enhance the performance of deep subspace clustering, extensive research has been conducted from various perspectives, primarily focusing on the three aspects, i.e., prior knowledge of the self-expression coefficient matrix [4], [21], [22], [23], [35], [36], [37], sample relations [38], feature fusion [39], [40], [41], [42], [43], and self-supervised learning [25], [26], [44], [45], [46].

Prior knowledge of the self-expression coefficient matrix has been explored in deep subspace clustering research. For instance, Peng et al. [21] proposed a deep subspace clustering framework called PARTY, which utilized hand-crafted features and linearly reconstructed data in the input space. The PARTY framework incorporated sparse coefficients from the original space into the latent space. In a similar vein, DSC-Net [23] employed a stacked convolutional autoencoder with a plug-in self-expression model, enabling an end-to-end training approach. Subsequent studies have introduced various prior knowledge techniques, including latent distribution preservation [35] and information entropy [4], [36], to enhance the performance of deep subspace clustering.

Feature fusion is crucial in deep subspace clustering. Valanarasu et al. [39] introduced a multi-layer feature fusion strategy to integrate latent features from various layers, boosting the descriptive power of deep features. Dang et al. [40] proposed a method that merges self-expression coefficients from multi-level features, using similarity constraints to create a strong affinity matrix. Building on this, Abavisani et al. [41] applied an adaptive augmentation technique for subspace clustering to improve deep subspace clustering by developing data augmentation strategies.

Self-supervised learning techniques have gained significant attention in deep subspace clustering. These methods leverage auxiliary tasks to extract supervisory information from the data, enabling the learning of useful representations for downstream tasks. Approaches like generative adversarial learning [25], [44] and pseudo label learning [26], [45] are commonly used. Zhou et al. [25] proposed an autoencoder-based method that generates subspaces through adversarial learning. Zhang et al. [26] incorporated self-supervised learning into an autoencoder-based subspace clustering framework and optimized network parameters using pseudo labels, leading to more discriminative deep features.

In contrast to autoencoder-based deep subspace clustering approaches that process entire datasets as a single batch, the proposed DISSC framework adopts mini-batches as inputs for deep neural networks. This is achieved by integrating contrastive learning and knowledge self-distillation principles into a deep subspace clustering model, making it suitable for large-scale data. Recently, many researchers have focused on the scalability problem in deep subspace clustering [47], [48], [49], [50]. Among them, NCSC [47] and SSCN [50] address scalability concerns without compromising clustering performance, relying on the consensus encoder-decoder paradigm, which may be susceptible to the aforementioned issues.

C. Knowledge Distillation

Knowledge distillation has been previously proposed to compress ensembles of deep neural networks. By leveraging teacher knowledge and employing distillation strategies in teacher-student learning, one can achieve effective performance using a lightweight student model. The learning schemes of knowledge distillation can be categorized into three main categories: offline distillation, online distillation, and self-distillation [51], depending on whether the teacher model is updated simultaneously with the student model. Self-distillation, as a special type of knowledge distillation, enhances the training of a student network by utilizing its own knowledge, without relying on a teacher network. It has shown significant superiority in practice.

In this work, the self-distillation model or the knowledge distillation framework is not directly employ. Instead, this paper draws inspiration from the concept of self-distillation and leverage valuable knowledge obtained from the network model itself, independent on a teacher model, to enhance feature learning performance. Furthermore, a relationship between self-distillation and subspace clustering is established, showing that self-distillation is integral in accurately predicting subspace assignments without relying on ground-truth labels and extracting meaningful subspace information from batch-by-batch data.

III. Methodology

A. Overview and Motivation

In this study, the proposed DISSC framework comprises three modules illustrated in Fig. 2: deep feature extraction, self-expression-based subspace learning, and inductive subspace clustering.

To achieve scalable subspace clustering, the dataset is divided into multiple mini-batches randomly. Within the deep feature extraction module, deep features are extracted from the current batch data and its counterparts using data augmentation through a weight-sharing neural network. Subsequently, the self-expression-based subspace learning module and the inductive subspace clustering module are developed to capture the affinities of data points from different perspectives. The former module enforces the self-expression property in a batch-by-batch manner, while the latter formulates subspace clustering as a multi-class prediction problem. These modules allow our framework to be applied to datasets of various sizes without relying on ad-hoc post-processing techniques.

To address the challenge of learning meaningful subspace information and achieving reliable subspace membership prediction without ground-truth labels, a novel nonlocal contrastive loss with self-distillation is proposed, which offers two key advantages as follows:

- Nonlocal contrastive learning enables the extraction of robust neighborhood information, improving the affinity matrices derived from the mini-batches in the two modules.
- The affinity matrices obtained from the two modules contain different supervision information. By introducing self-distillation, the proposed DISSC collaboratively extracts complementary knowledge from these affinity matrices. This means that the distilled knowledge from the self-expression based subspace learning module can be incorporated into the inductive subspace clustering module, and vice versa.

Recently, contrastive learning has been explored as a solution to clustering problems in studies such as [52], [53], [54] due to its powerful representation learning capabilities. Although the Multi-view Contrastive Graph Clustering (MCGC) method

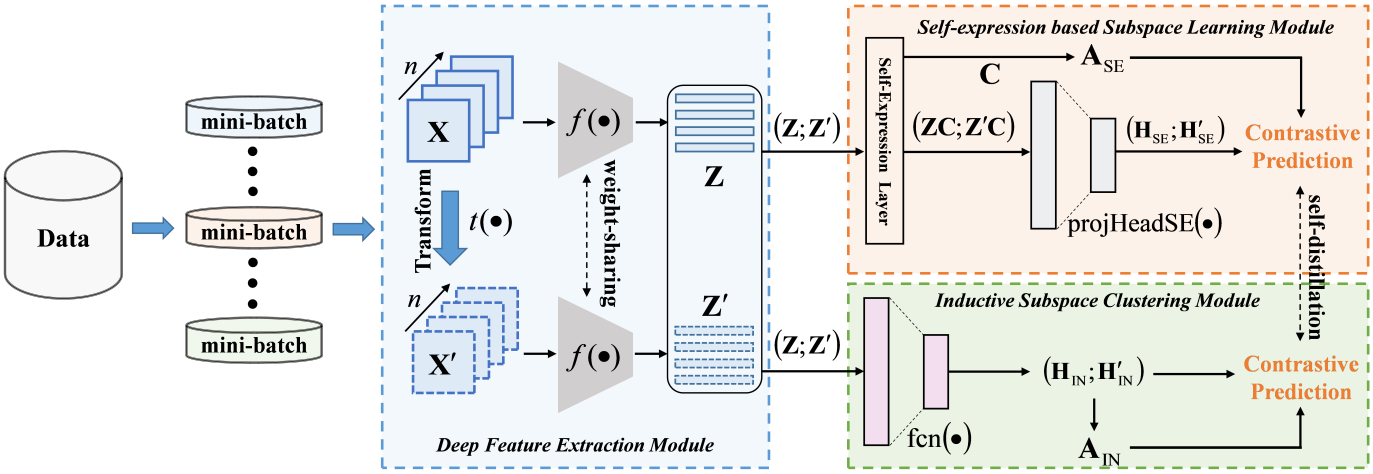


Fig. 2. The network architecture of proposed DISSC framework. During the training process, the large-scale data can be sampled randomly into mini-batches. Each mini-batch data undergoes three main modules: (1) deep feature extraction, where the mini-batch data $\boldsymbol{X}$ is augmented to produce its counterpart $\boldsymbol{X}'$. The weight-sharing convolutional encoding networks are then used to capture the collection of original and transformed features $\{\boldsymbol{Z}; \boldsymbol{Z}'\}$ for downstream tasks; (2) self-expression based subspace learning, in which the self-expression matrix $\boldsymbol{C}$ can be used to generate the affinity matrix $\boldsymbol{A}_{\mathrm{SE}}$; (3) inductive subspace clustering, where the affinity matrix $\boldsymbol{A}_{\mathrm{IN}}$ is achieved by calculating the similarities between these embeddings via $\mathtt{fcn}(\cdot)$. Additionally, self-distillation is utilized to regularize the contrastive prediction in both the two modules, fostering a closed-loop training approach.

proposed in [53] applies contrastive learning to the self-expressive matrix, our DISSC method differs significantly from MCGC. Specifically, MCGC uses a contrastive penalty on a single affinity matrix to reduce negative similarities, relying on k-nearest neighbors for each data view. In contrast, our DISSC method decouples affinity computation into two separate affinity matrices and applies a nonlocal contrastive loss between them. Additionally, this paper introduces a self-distillation mechanism that improves the self-expressive matrix by leveraging both positive and negative supervision from the two affinity matrices.

B. Computation of Affinity Matrices

1) Deep feature extraction: As illustrated in Fig. 2, the deep feature extraction module aims to capture intermediate deep features of the original image data $\mathcal{X} \in \mathbb{R}^{w\times h\times n}$ and its counterpart $\mathcal{X}' \in \mathbb{R}^{w\times h\times n}$ within each mini-batch $\mathcal{B}$, where n is the batch size, i.e., $|\mathcal{B}| = n$. For simplicity, let $\mathcal{X} = \{\mathcal{X}^i\}_{i=1,2,\cdots,n}$ denote the set of original image data and $\mathcal{X}' = \{\mathcal{X}^{n+i}\}_{i=1,2,\cdots,n}$ represent the set of transformed image data obtained through the predefined data transformation function $\mathtt{transform}(\cdot)$: $\mathcal{X}^{n+i} = \mathtt{transform}(\mathcal{X}^i)$. Utilizing a weight-sharing convolutional encoding network denoted as $\mathtt{convEnc}(\cdot)$, the intermediate representations can be expressed as $\boldsymbol{Z} = \mathtt{convEnc}(\mathcal{X}) \in \mathbb{R}^{d\times n}$ and $\boldsymbol{Z}' = \mathtt{convEnc}(\mathcal{X}') \in \mathbb{R}^{d\times n}$, respectively, where d is the feature dimension. A range of transformation operations can be empirically applied as part of the data augmentation policies, such as random color distortion (sharpness), random clip and rotation, and random Gaussian blur. These operations are commonly adopted as routine pre-processing strategies. $\boldsymbol{Z} = \mathtt{convEnc}(\mathcal{X}) \in \mathbb{R}^{d\times n}$ and $\boldsymbol{Z}' = \mathtt{convEnc}(\mathcal{X}') \in \mathbb{R}^{d\times n}$, respectively, where d is the feature dimension. A range of transformation operations can be empirically applied as part of the data augmentation policies, such as random color distortion (sharpness), random clip and rotation, and random Gaussian blur. These operations are commonly adopted as routine pre-processing strategies.

2) Self-expression based subspace learning: The self-expression property, which characterizes a data point drawn from linear subspaces as a linear combination of other data points within the same subspace, has demonstrated effectiveness in subspace clustering according to existing literature. Despite the significant appearance variations between the original data and its transformed counterpart, this property encourages the self-expression matrix to acquire consistent subspace information. To achieve this, a shared self-expression layer is introduced, parameterized as $\boldsymbol{C} \in \mathbb{R}^{n\times n}$, leading to:

$$\mathcal{L}_{\mathrm{SE}} = \frac{1}{2}\left\|\begin{bmatrix}\boldsymbol{Z}\\ \boldsymbol{Z}'\end{bmatrix} - \begin{bmatrix}\boldsymbol{Z}\\ \boldsymbol{Z}'\end{bmatrix}\boldsymbol{C}\right\|_{\mathrm{F}}^2 + \frac{\alpha}{2}\|\boldsymbol{C}\|_{\mathrm{F}}^2, \tag{3}$$

where $\begin{bmatrix}\boldsymbol{Z}\\ \boldsymbol{Z}'\end{bmatrix} \in \mathbb{R}^{2d\times n}$ indicates a union of the encoded features corresponding to original and augmented data and $\alpha > 0$ is a trade-off parameter ($\alpha = 1$ in the experiments). The self-expression layer, as depicted in Eqn. (3), is designed to be shared by both the original data and augmented data. This shared layer ensures an invariant subspace-preserving property, even under random transformations applied to the input data. By using a single coefficient matrix to represent the weights of the shared self-expression layer, this module estimates a consistent affinity between different views of the same data. This estimation leads to the formation of the affinity matrix in the self-expression based subspace learning module $\boldsymbol{A}_{\mathrm{SE}} \in [0,1]^{2n\times 2n}$, which is partitioned into four $n\times n$ blocks. Notably, each block is identical to the matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ and can be formulated as follows:

$$\boldsymbol{A}_{\mathrm{SE}}=\begin{bmatrix}\boldsymbol{A} & \cdots & \boldsymbol{A}\\ \vdots & \ddots & \vdots\\ \boldsymbol{A} & \cdots & \boldsymbol{A}\end{bmatrix},\quad \boldsymbol{A}\left(i,j\right)=\begin{cases}\frac{|\boldsymbol{C}(i,j)|+|\boldsymbol{C}(j,i)|}{2a_{\max}}, & i\neq j\\ 1, & i=j\end{cases}, \tag{4}$$

where $a_{\max}$ is the maximum absolute value of off-diagonal entries of the current row.

It can be inferred from Eqn. (4) that $\boldsymbol{A}_{\mathrm{SE}}$ solely captures the similarities within the limited data present in the current batch, whereas accurate similarities of the entire dataset are crucial for obtaining reliable clustering results. To address this issue, a nonlocal contrastive loss with self-distillation is proposed to enhance the affinity matrix. In the preparation for the loss function, instead of utilizing the commonly used data reconstruction task, this paper introduces a nonlocal contrastive prediction task within this module. This task aims to learn subspace-preserving representations that enable informative data affinity. Following the approach [55], a two-layer projection head $\mathtt{projHeadSE}(\cdot)$ is designed to nonlinearly map the intermediate representations into another space, thereby avoiding information loss caused by contrastive learning.

$$\begin{aligned}\mathbf{H}_{\mathrm{SE}}&=\mathtt{projHeadSE}\left(\begin{bmatrix}\boldsymbol{Z}\\ \boldsymbol{Z}'\end{bmatrix}\boldsymbol{C}\right)\\ &=\left[\underbrace{\mathbf{H}_{\mathrm{SE}}^{1},\mathbf{H}_{\mathrm{SE}}^{2},\cdots,\mathbf{H}_{\mathrm{SE}}^{n}}_{\mathtt{projHeadSE}(\boldsymbol{ZC})},\underbrace{\mathbf{H}_{\mathrm{SE}}^{n+1},\mathbf{H}_{\mathrm{SE}}^{n+2},\cdots,\mathbf{H}_{\mathrm{SE}}^{2n}}_{\mathtt{projHeadSE}(\boldsymbol{Z}'\boldsymbol{C})}\right].\end{aligned} \tag{5}$$

3) Inductive subspace clustering: It is widely acknowledged that data belonging to the same subspace should have identical ground-truth class labels. Hence, the segmentation of data into separate subspaces can be formulated as a classification problem. Our primary goal is to acquire discriminative label features that serve as representation features during training and as one-hot subspace indices during testing. Hence, one can effectively learn and distinguish between different subspaces within the dataset while preserving their intrinsic characteristics. To this end, a subspace classifier $\mathtt{fcn}(\cdot)$ which comprises two fully connected layers with ReLU and Softmax respectively is designed for nonlinear activation, and the cluster indicator representations can be achieved as follows:

$$\begin{aligned}\mathbf{H}_{\mathrm{IN}}&=\mathtt{fcn}\left(\begin{bmatrix}\boldsymbol{Z}\\ \boldsymbol{Z}'\end{bmatrix}\right)\\ &=\left[\underbrace{\mathbf{H}_{\mathrm{IN}}^{1},\mathbf{H}_{\mathrm{IN}}^{2},\cdots,\mathbf{H}_{\mathrm{IN}}^{n}}_{\mathtt{fcn}(\boldsymbol{Z})},\underbrace{\mathbf{H}_{\mathrm{IN}}^{n+1},\mathbf{H}_{\mathrm{IN}}^{n+2},\cdots,\mathbf{H}_{\mathrm{IN}}^{2n}}_{\mathtt{fcn}(\boldsymbol{Z}')}\right].\end{aligned} \tag{6}$$

Besides, a nonlocal contrastive loss is proposed as the pretext objective function to indirectly impose requirements for good subspace structures. Similarly, given the output feature matrix $\mathbf{H}_{\mathrm{IN}}$, an affinity matrix $\boldsymbol{A}_{\mathrm{IN}}\in[0,1]^{2n\times 2n}$ is constructed via:

$$\boldsymbol{A}_{\mathrm{IN}}\left(i,j\right)=\mathtt{sim}\left(\mathbf{H}_{\mathrm{IN}}^{\mathtt{i}},\mathbf{H}_{\mathrm{IN}}^{\mathtt{j}}\right), \tag{7}$$

where $\mathtt{sim}(\boldsymbol{a},\boldsymbol{b})=\frac{\boldsymbol{a}\boldsymbol{b}^{\mathcal{T}}}{\|\boldsymbol{a}\|\cdot\|\boldsymbol{b}\|}$ represents the pair-wise similarity measured by cosine distance.

C. Nonlocal Contrastive Learning with Self-Distillation

Knowledge distillation was firstly introduced for model compression where a student network is trained to mimic the behavior of a teacher network. The recent success of self-distillation shows the feasibility that one network can enjoy an obvious enhancement to its performance by teaching itself. The idea of using self-distillation for subspace clustering is simple yet effective. It can be observed that the proposed self-expression based subspace learning and inductive subspace clustering modules produce affinity matrices from two different perspectives, yet based on the same data samples. Since the proposed DISSC model estimates data affinities from these complementary perspectives, it naturally distills knowledge from one affinity matrix to the other. More precisely, the model distills its own output to extract so-called "dark knowledge" from one perspective, which serves as auxiliary supervision for the other perspective, guided by the two observations as follows:

- The data affinities $\boldsymbol{A}_{\mathrm{SE}}(i,j)$ derived from the self-expression layer are ideally nonzero if and only if $\mathcal{X}^i$ and $\mathcal{X}^j$ come from the same subspace and therefore powerful to reflect the intrinsic similarity among intra-subspace data, it is reasonable to take full advantage of the subspace information in this affinity matrix $\boldsymbol{A}_{\mathrm{SE}}$ to mine positive supervision for the contrastive learning, escaping from the predicament of local positive pair searching.
- The inductive subspace clustering module also estimates pairwise affinities based on the data distribution in the latent feature space, the consequent affinity matrix helps to reveal the inherent relation among inter-subspace data. That is, the data affinities $\boldsymbol{A}_{\mathrm{IN}}(i,j)$ would be close to zero if $\mathcal{X}^i$ and $\mathcal{X}^j$ highly likely to lie in different subspaces. Therefore, the affinity matrix $\boldsymbol{A}_{\mathrm{IN}}$ can be leveraged to explore negative supervision to facilitate the nonlocal contrastive prediction task.

As discussed, the nonlocal contrastive loss can be designed by distilling the complementary knowledge and transferring it from one affinity to another. It's well-known that contrastive learning aims to maximize similarities of positive pairs while minimizing those of negative ones. The objective of the nonlocal contrastive learning can be formulated as the maximization of the mutual information between data and the corresponding representations by drawing them up closer to their reliable positive samples and differentiating them against their potential negative samples. Given one data point $\mathcal{X}^i$, the possibility that $\mathcal{X}^i$ and $\mathcal{X}^j$ are recognized to lie in the same subspace can be formulated as:

$$\begin{aligned}P_{\mathrm{SE}}\left(i,j\right)&=\frac{\exp\left(\mathrm{sim}\left(\mathbf{H}_{\mathrm{SE}}^{i},\mathbf{H}_{\mathrm{SE}}^{j}\right)/\tau\right)}{\sum_{k=1}^{2n}\exp\left(\mathrm{sim}\left(\mathbf{H}_{\mathrm{SE}}^{i},\mathbf{H}_{\mathrm{SE}}^{k}\right)/\tau\right)},\\ P_{\mathrm{IN}}\left(i,j\right)&=\frac{\exp\left(\mathrm{sim}\left(\mathbf{H}_{\mathrm{IN}}^{i},\mathbf{H}_{\mathrm{IN}}^{j}\right)/\tau\right)}{\sum_{k=1}^{2n}\exp\left(\mathrm{sim}\left(\mathbf{H}_{\mathrm{IN}}^{i},\mathbf{H}_{\mathrm{IN}}^{k}\right)/\tau\right)},\end{aligned} \tag{8}$$

where τ is the temperature parameter that controls the concentration level of the distribution in the two modules.

Correspondingly, the probabilities of $\mathcal{X}^i$ and $\mathcal{X}^j$ not being recognized to be the same subspace are $1 - P_{\text{SE}}(i,j)$ and $1 - P_{\text{IN}}(i,j)$ in the self-expression based subspace learning and inductive subspace clustering modules, respectively. To comprehensively explore the useful knowledge from different affinities, the positive-pair knowledge can be captured from $\boldsymbol{A}_{\text{SE}}$ while the negative-pair knowledge can be achieved from $\boldsymbol{A}_{\text{IN}}$. Assuming that different data point pairs being classified into the same subspace are independent, the joint probabilities of one arbitrary data point $\mathcal{X}^i$ with **Self-Distillation Regularization (SDR)** in the two modules are defined as:

$$P_{\text{SE}}\left(\mathcal{X}^i\right) = \prod_{j\in\mathcal{P}(i)} P_{\text{SE}}(i,j) \underbrace{\prod_{j\in\mathcal{N}(i)} \left(1 - P_{\text{SE}}(i,j)\right)}_{\text{SDR}}, \quad (9)$$

$$P_{\text{IN}}\left(\mathcal{X}^i\right) = \underbrace{\prod_{j\in\mathcal{P}(i)} P_{\text{IN}}(i,j)}_{\text{SDR}} \prod_{j\in\mathcal{N}(i)} \left(1 - P_{\text{IN}}(i,j)\right), \quad (10)$$

where $\mathcal{P}(i) = \{j|\boldsymbol{A}_{\text{SE}}(i,j) \geq \xi\}$, $\mathcal{N}(i) = \{j|\boldsymbol{A}_{\text{IN}}(i,j) \leq 1-\xi\}$ and $\xi \in (0,1)$ is a thresholding parameter to determine the reliability of positive or negative samples respectively. Specifically, the first part of Eq. (9) utilizes positive supervision from $\mathcal{P}(i)$ by employing the probability $P_{\text{SE}}(i,j)$ to indicate that $\mathcal{X}^i$ and $\mathcal{X}^j$ belong to the same subspace. The latter part provides negative supervision from $\mathcal{N}(i)$ by using the probability $1-P_{\text{SE}}(i,j)$ to indicate that $\mathcal{X}^i$ and $\mathcal{X}^j$ do not belong to the same subspace. A similar approach is applied in Eq. (10). By distilling knowledge from the model itself, SDR enhances the affinity learning via extra knowledge distilled from one module to another.

Compared to traditional contrastive learning, the proposed nonlocal contrastive learning method relies on the information of positive and negative samples to enhance the clustering-friendly representation learning ability. The self-distillation regularization enables our framework to fall into a virtuous circle, the self-expression-based subspace learning module learns more subspace-revealing representations for affinity measurement and provides refined supervision for the inductive subspace clustering module in return, and vice versa.

The loss function of nonlocal contrastive learning with self-distillation regularization aims to minimize the sum of the negative log-likelihood over all the data points within the current batch, combining Eqns. (9) and (10) as:

$$\mathcal{L}_{\text{SDR}} = -\sum_i \log P_{\text{SE}}\left(\mathcal{X}^i\right) - \sum_i \log P_{\text{IN}}\left(\mathcal{X}^i\right). \quad (11)$$

D. Model Training

To obtain an end-to-end trainable framework, an overall loss function is designed by putting together the losses defined in Eqns. (3) and (11), leading to:

$$\mathcal{L} = \mathcal{L}_{\text{SDR}} + \lambda\mathcal{L}_{\text{SE}}. \quad (12)$$

Algorithm 1 Deep Inductive and Scalable subspace clustering via nonlocal constrastive self-distillation

Input: Original data $\mathcal{X}$, tradeoff parameters λ, temperature factor τ, augmentation family $t(\cdot)$, thresholding parameters ξ, maximum iteration M.
 # *Training Stage*
Pre-train the network via Eqn. (12) by fixing $\boldsymbol{A}_{\text{SE}} = \mathbf{I}$, $\boldsymbol{A}_{\text{IN}} = \mathbf{I}$, and $\boldsymbol{C} = \mathbf{I}$ to achieve the encoding parameters $f(\cdot)$.
for $iteration = 1$ **to** M **do**
  Randomly sample a mini-batch data.
  Forward the batch data through the encoder and subspace generator to update the self-expression layer via Eqn. (3).
  Jointly update all the parameters by minimizing the loss function defined in Eqn. (12).
end for
 # *Testing Stage*
for each data point $\boldsymbol{x}_i$, $i = 1$ **to** n **do**
  Forward it through the inductive subspace clustering module and infer the cluster label via Eqn. (13).
end for
Output: The cluster assignment for all samples.

Note that for easy to tune hyperparameters, as shown in Eqn. (8), the temperature parameter τ is shared by the subspace learning and inductive subspace clustering modules.

1) Pre-training stage: At specific implementation, the network is pre-trained by fixing the parameters of the self-expression layer as an identity matrix, i.e., $\boldsymbol{C} = \mathbf{I}$, which means that each data point is attached with a different subspace membership at the beginning. Since the self-expression layer is frozen, the network is trained without the self-expression layer. Therefore, the pre-training stage essentially trains an instance-level contrastive learning model to achieve meaningful latent features for yielding reliable affinity matrices $\boldsymbol{A}_{\text{SE}}$, $\boldsymbol{A}_{\text{IN}}$ at the pre-training stage.

2) Fine-tuning stage: After the pre-training stage, our framework has been equipped with preliminary discriminative ability. In the fine-tuning stage, for every randomly sampled mini-batch data, a mini-batch data is fed into the subspace learning module and update the self-expression layer via Eqn. (3) for initialization. Subsequently, all of the parameters can be jointly updated in the proposed DISSC framework with Eqn. (12) until the maximum epochs are reached. In the testing phase, it is needless to infer cluster labels via the parameters of the self-expression layer. Instead, the cluster assignment can be computed via Eqn. (13).

$$c = \arg\max_i(\mathbf{H}^i_{\text{IN}}), \quad (13)$$

where c indicates the subspace the data point $\boldsymbol{x}_i$ is mostly likely to be in. The full algorithm of the proposed framework, including the training stage and testing stage, is summarized in Algorithm 1.

IV. Experiment

In this section, extensive experiments are conducted on large-scale benchmark datasets to comprehensively evaluate the performance of the proposed method in the deep subspace clustering task. Experiments are conducted on the same data with the same experimental settings for a fair comparison. It is important to note that `large-scale data` does not have a universally defined threshold. However, in our experiments, all the datasets utilized can be considered relatively larger in scale when compared to those commonly used in the deep subspace clustering community, such as Deep Subspace Clustering Networks [23] and its variants [25], [26], [35], [39], [40], [41], [44], [45], which rely on convolutional autoencoder backbones.

A. Datasets

1) MNIST: MNIST[1] dataset consists of up to 70000 grayscale images of hand-written digits ranging from zero to nine, each of which is resized into 28×28. It has been widely used in evaluating the performance in the pattern recognition task due to the challenging variations of the digits. Generally, MNIST can be divided into MNIST-train and MNIST-test, which includes 60,000 and 10,000 images respectively. In this experiment, all of the images from MNIST dataset are used for clustering.

2) Fashion-MNIST: Fashion-MNIST[2] is a dataset comprising of 28×28 grayscale images of 70,000 fashion products corresponding to 10 categories varies in their styles with different consumer groups. Similar to MNIST, Fashion-MNIST-full is split into Fashion-MNIST-train and Fashion-MNIST-test which also consist of 60,000 and 10,000 images for each. Similar with the settings on MNIST dataset, the entire set of images is employed for evaluating the performance of deep subspace clustering.

3) CIFAR-10: The CIFAR-10[3] dataset is a widely used benchmark in machine learning, particularly for image classification tasks. It contains 60,000 32×32 color images, distributed across 10 classes: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck. There are 6,000 images per class, with 50,000 images for training and 10,000 images for testing. The entire set of images is employed for evaluating the performance of deep subspace clustering.

4) STL-10: The STL-10[4] dataset is a collection of images used primarily for evaluating machine learning algorithms, especially in the context of image classification. It contains 10 classes, with 5,000 training images and 8,000 test images, each with a resolution of 96×96 pixels in color. The dataset includes a variety of natural scenes, such as airplanes, cars, and animals. STL-10 is designed to be more challenging than similar datasets like CIFAR-10, offering higher resolution images and a more diverse set of objects.

5) Stanford Online Products: Stanford Online Products(SOP)[5] dataset consists of images belonging to 10 classes excluding kettle and lamp, approximately 1,000 images per category, are selected and then down-sampled to 32×32 gray images. The SOP dataset is more challenging than MNIST and Fashion-MNIST due to greater variations in samples and is commonly used in metric learning tasks. In this experiment, the same settings in [47] are used for comparison.

B. Experiment Implementation

In the experiments, batch normalization is not used in the framework to avoid corrupting the subspace structure intended to be learned in the latent space. The same encoder network architecture as in [47] is used for feature extraction on the MNIST, Fashion-MNIST, and SOP datasets, and the architecture from [56] is applied to the STL-10 and CIFAR-10 datasets to ensure a fair comparison. For the subspace learning module, the dimensionality of the output feature space is empirically set as 128 to keep sufficient information of images. As for the inductive subspace clustering module, the output feature dimensionality is defined as the number of subspaces. The Adam optimizer with an initial learning rate of 1×10^{-3} is adopted to optimize the entire network without weight decay and scheduler. The batch size varies in different datasets and will be given in the experimental results on the datasets. The threshold parameter ξ is 0.8 for MNIST and slightly changed for other datasets.

An augmentation policy is defined as a function $\texttt{transform}(\cdot)$ to map an image $\boldsymbol{x}$ to an augmented image given a discrete strength parameter m. Note that the strength parameter is not used by all augmentations, but some use it to define how strongly to distort the image. The choice of augmentation policy plays an important role in the performance of the network. Following [57], an image $\boldsymbol{x}$ and an augmentation space S are taken as input. An augmented image can be obtained by simply sampling an augmentation from S uniformly at random and applying this augmentation on the given image $\boldsymbol{x}$ with a strength m, sampled uniformly at random from the set of possible strengths $\{0, 1, 2, \cdots, N\}$. Unlike [41], which generates a complex data distribution through multiple sequential augmentations, this approach focuses on providing a data distribution based on the mean of the views on the data through a variety of single augmentations. This parameter-free strategy can be elegantly applied to large-scale datasets without incurring heavy computational costs associated with searching for specific augmentations. The augmentation space consists of Identity mapping, FlipLR, FlipUD, ShearX, ShearY, Posterize, Rotate, Invert, Brightness, Equalize, Solarize, Contrast, AutoContrast, Sharpness, TranslateX, TranslateY, Cutout. The magnitude range is the same as in [41] and is discretized into 30 values. It is worth noting that FlipLR, FlipUD and Cutout are excluded from the full space for MNIST.

[1] http://yann.lecun.com/exdb/mnist/
[2] https://github.com/zalandoresearch/fashion-mnist
[3] http://www.cs.toronto.edu/~kriz/cifar.html
[4] https://cs.stanford.edu/~acoates/stl10/
[5] https://cvgl.stanford.edu/projects/lifted_struct/

C. Evaluation Metrics

In the experiments, several widely-used metrics are employed to evaluate the clustering performance, including clustering accuracy (ACC), normalized mutual information (NMI), and adjusted Rand index (ARI). These metrics [58] have been extensively utilized in the literature.

- ACC is employed to measure the hit rate in the classification task. On the other hand, when it comes to the clustering task, ACC can be defined as follows:

$$\mathrm{ACC} = \frac{1}{n}\sum_{i=1}^{n}\psi\left(l_i, bestmap(p_i)\right), \tag{14}$$

where l_i and p_i represent the true and predicted labels of the i-th image sample, respectively, and $\psi(X, Y)$ denotes the relationship between a and b:

$$\psi(X,Y) = \begin{cases} 1, & X = Y \\ 0, & X \neq Y \end{cases} \tag{15}$$

To address the issue of random label allocation in the clustering results, the function $bestmap(\cdot)$ [6] is utilized, which is solved using the well-known Kuhn-Munkres algorithm [59]. This function helps to establish the correspondence between the true labels and the predicted labels.

- NMI, a measure derived from information theory, compares two overlapping clusters. Unlike ACC, NMI quantifies the similarity between two clustering results and remains unaffected by the arrangement of cluster labels:

$$\mathrm{NMI}(X,Y) = \frac{MI(X,Y)}{\max(H(X), H(Y))}, \tag{16}$$

where $MI(X, Y)$ is the mutual information between X and Y, and $H(X)$ and $H(Y)$ are the entropies of X and Y, respectively.

- ARI assesses the similarity between two clusterings, while taking into account the possibility of chance agreement. ARI is defined as follows:

$$\mathrm{ARI}(X,Y) = \frac{(\mathrm{RI} - \mathbb{E}(\mathrm{RI}))}{(\max(\mathrm{RI}) - \mathbb{E}(\mathrm{RI}))}, \tag{17}$$

where RI is the observed Rand Index of the two clusterings, $\mathbb{E}(\mathrm{RI})$ is the expected value of Rand Index under a null hypothesis of random labeling, and $\max(\mathrm{RI})$ is the maximum possible value of Rand Index while ignoring permutations of the cluster labels.

It is obvious that ACC and NMI take a value within the interval $[0, 1.0]$, and the ARI ranges from -1 to 1, with 1 indicating perfect agreement between the two clusterings and 0 or negative values indicating the presence of randomness. Overall, larger values of ACC, NMI, and ARI indicate better clustering performance.

[6]http://www.cad.zju.edu.cn/home/dengcai/Data/Clustering.html

D. Compared Algorithms

To provide a comprehensive evaluation of the proposed algorithm against various state-of-the-art clustering methods, a selection of representative algorithms from diverse research areas is included. Different comparison methods are chosen for each dataset to ensure a fair evaluation, as these methods are typically optimized for datasets with distinct characteristics. Additionally, due to the non-uniformity of the datasets used, it was not possible to find identical comparison methods across all selected datasets. These competitive algorithms include:

1) Deep clustering approaches without subspace learning: To demonstrate the subspace clustering performance, state-of-the-art deep clustering approaches without subspace learning in their frameworks are used, including Deep Embedded Clustering (DEC) [60], Improved Deep Embedded Clustering (IDEC) [61], Joint Unsupervised LEarning of deep representations and image clusters (JULE) [62], Deep Adaptive image Clustering (DAC) [63], Deep Clustering Network (DCN) [64], DEeP Embedded RegularIzed ClusTering (DEPICT) [65], Latent space clustering in Generative Adversarial Networks (ClusterGAN) [66], deep clustering based on PartItion Confidence mAximisation (PICA) [67], Deep Clustering via Contractive Feature representation and focal loss [68], VanGAN-GMM [69], Local-to-Global deep Clustering on Approximate Uniform Manifold (LGC-AUM) [70], and Wasserstein Embedding Clustering (WEC) [71].

2) Scalable subspace clustering: In order to handle the large-scale datasets used in our experiments, several scalable subspace clustering methods are utilized without involving neural networks, such as Elastic-Net Subspace Clustering (EnSC) [32], Sparse Subspace Clustering by Orthogonal Matching Pursuit (SSC-OMP) [33], and scalable Exemplar-based Subspace Clustering (ESC) [72].

3) Deep autoencoder-based subspace clustering: Although there are numerous deep autoencoder-based subspace clustering methods available, experiments involving these algorithms were not conducted on the large-scale datasets used in this work, likely due to computational constraints. As a result, two-step methods are incorporated for comparison, where intermediate features are learned using Convolutional AutoEncoders (CAE), followed by subspace clustering. Specifically, methods such as Low-Rank Representation (LRR)+CAE [23], Sparse Subspace Clustering (SSC)+CAE [23], Kernel Sparse Subspace Clustering (KSSC)+CAE [23], Efficient Deep Embedded Subspace Clustering (EDESC) [73], and Adaptive Graph Convolutional Subspace Clustering (AGCSC) [74] are compared in the experiment.

4) Scalable deep subspace clustering: In addressing the challenge of deep subspace clustering on large-scale datasets, the proposed algorithm is compared with state-of-the-art scalable methods introduced in recent years, including scalable deep k-subspace clustering (DKSC) [49], Neural Collaborative Subspace Clustering (NCSC) [47], Self-Expressive Network (SENet) [48], Few-Shot Subspace Clustering Learning (FS2CL) [56], and Very Deep Representation Learning for Subspace Clustering (VDRL-SC) [75], all of which have demonstrated outstanding performance in this domain. These

TABLE II
CLUSTERING RESULTS (IN %) IN TERMS OF ACC, NMI, AND ARI USING COMPETITIVE ALGORITHMS ON MNIST, FASHION-MNIST, CIFAR-10, STL-10, AND SOP DATASETS. BEST (SECOND) RESULTS ARE HIGHLIGHTED IN BOLD (UNDERLINE).

Methods	Reference	MNIST			Fashion-MNIST			CIFAR-10			STL-10			SOP		
		ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
DEC [60]	ICML 2016	84.3	80.0	75.0	59.0	60.1	44.6	30.1	25.7	16.1	35.9	27.6	18.6	22.9	12.1	3.6
JULE [62]	CVPR 2016	96.4	91.3	92.7	56.3	60.8	N/A	27.2	19.2	13.8	27.7	18.2	16.4	N/A	N/A	N/A
SSC-OMP [33]	CVPR 2016	92.8	84.2	84.9	27.4	42.1	19.6	32.6	49.8	N/A	22.1	13.6	11.5	13.2	2.6	1.4
EnSC [32]	CVPR 2016	98.0	94.5	**95.7**	67.2	70.5	56.5	N/A	N/A	N/A	61.3	60.1	N/A	N/A	N/A	N/A
IDEC [61]	IJCAI 2017	88.1	86.7	76.3	52.9	55.7	49.6	23.5	10.4	N/A	37.8	32.5	N/A	N/A	N/A	N/A
DAC [63]	ICCV 2017	97.8	93.8	94.9	61.5	63.2	50.2	52.2	39.6	30.6	47.0	36.6	25.7	23.1	9.8	6.2
DCN [64]	ICML 2017	83.3	80.9	74.9	58.7	59.4	43.0	N/A	N/A	N/A	N/A	N/A	N/A	21.3	8.4	3.1
DSC-Net [23]	NeurIPS 2017	65.9	73.0	57.1	60.6	61.7	48.2	N/A	N/A	N/A	N/A	N/A	N/A	26.9	14.6	8.8
SSC+CAE [23]	NeurIPS 2017	43.0	56.8	28.6	35.9	18.1	13.5	19.8	7.8	6.9	23.2	15.3	16.1	12.7	0.7	0.2
LRR+CAE [23]	NeurIPS 2017	55.2	66.5	40.6	34.5	25.4	10.3	21.5	8.2	7.7	25.8	16.8	16.2	22.4	17.4	4.0
KSSC+CAE [23]	NeurIPS 2017	58.5	67.7	49.4	38.2	19.7	14.7	22.1	11.2	10.5	26.1	15.7	16.1	26.8	15.2	7.5
DEPICT [65]	ICCV 2017	96.5	91.7	N/A	39.2	39.2	N/A	32.6	27.4	N/A	37.1	30.3	N/A	N/A	N/A	N/A
DKSC [49]	ACCV 2018	87.1	78.2	75.8	63.8	62.0	48.0	N/A	N/A	N/A	N/A	N/A	N/A	22.9	16.6	7.2
ESC [72]	ECCV 2018	97.1	92.5	93.6	66.8	70.8	55.6	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
ClusterGAN [66]	AAAI 2019	95.0	89.0	89.0	63.0	64.0	50.0	41.2	32.3	N/A	53.6	60.6	59.3	N/A	N/A	N/A
NCSC [47]	ICML 2019	94.1	86.1	87.5	72.1	68.6	59.2	N/A	N/A	N/A	N/A	N/A	N/A	27.5	13.8	7.7
PICA [67]	CVPR 2020	94.1	89.8	89.5	64.1	63.4	54.8	69.6	59.1	51.2	71.3	61.1	53.1	N/A	N/A	N/A
SENet [48]	CVPR 2021	96.8	91.8	93.1	69.7	66.3	55.6	76.5	65.5	N/A	N/A	N/A	N/A	N/A	N/A	N/A
EDESC [73]	CVPR 2022	91.3	86.2	N/A	63.1	67.0	N/A	62.7	46.4	N/A	74.5	68.7	N/A	N/A	N/A	N/A
DCCF [68]	PR 2022	97.4	93.3	N/A	62.1	64.6	N/A	45.8	36.2	N/A	72.8	66.8	N/A	N/A	N/A	N/A
VanGAN-GMM [69]	TNNLS 2022	95.5	91.7	89.4	63.8	63.3	46.3	28.8	15.8	19.8	52.2	50.2	58.1	27.3	13.1	6.8
AGCSC [74]	CVPR 2023	95.6	92.3	90.1	60.6	60.9	49.3	54.5	41.4	38.7	54.4	51.6	49.2	22.6	14.2	7.9
LGC-AUM [70]	TKDE 2023	97.5	**94.9**	94.6	65.4	64.3	54.2	63.1	52.9	44.0	64.5	52.9	44.2	N/A	N/A	N/A
WEC [71]	TMM 2024	96.7	92.2	N/A	62.3	62.7	N/A	49.2	37.0	N/A	70.4	67.3	N/A	N/A	N/A	N/A
FS2CL [56]	TCSVT 2024	93.2	90.2	N/A	68.3	66.8	N/A	69.4	62.1	53.7	79.6	74.1	62.7	N/A	N/A	N/A
VDRL-SC [75]	TKDE 2024	N/A	N/A	N/A	74.9	67.9	60.6	64.6	62.9	45.7	N/A	N/A	N/A	N/A	N/A	N/A
DISSC	This paper	**98.3**	94.7	**95.7**	**80.5**	**74.5**	**62.0**	**77.2**	**66.3**	**54.2**	**80.1**	**75.6**	**63.9**	**41.6**	**27.5**	**17.6**

approaches serve as our related works for tackling the deep subspace clustering problem in the context of large-scale data.

E. Experimental Results

The experimental results in terms of ACC, NMI and ARI on the MNIST, Fashion-MNIST, CIFAR-10, STL-10 and Stanford Online Products datasets are reported in Table II. It is worth highlighting that the SOP dataset is seldom used in related works. Nevertheless, it is chosen to demonstrate the effectiveness of the proposed method. Given that the methods being compared differ across various datasets, some entries in the table are marked as 'N/A' to signify that the results are not available for the corresponding methods.

1) Experimental Results on MNIST and Fashion-MNIST Datasets: The results on MNIST dataset indicate that the autoencoder-based deep subspace clustering methods, i.e., DSC-Net, LRR+CAE, SSC+CAE, KSSC+CAE, exhibit comparatively lower performance in terms of ACC, NMI, and ARI when compared to other approaches. This observation suggests that the representation learning guided by pixel-level reconstruction from CAE is insufficient in capturing the underlying structural information effectively. Additionally, applying other state-of-the-art deep subspace clustering methods to large-scale datasets proves to be impractical. within the set of compared algorithms, EnSC exhibits outstanding performance in the domain of scalable subspace clustering. Moreover, in terms of the NMI metric, LGC-AUM [70] attains the top performance, while our DISSC secures the second-best ranking. In general, the proposed DISSC attains the optimal subspace clustering results in terms of the majority of metrics. This underlines the effectiveness of contrastive self-distillation within the framework of deep subspace clustering, further validating the superiority and innovation of the proposed DISSC algorithm in the field of subspace clustering. The overall clustering performance on Fashion-MNIST dataset is notably lower than that achieved on the MNIST dataset, primarily attributed to the inherent challenges posed by these product images from Fashion-MNIST dataset. Notably, SSC-OMP exhibits the poorest performance compared to the other evaluated methods. This disparity can be attributed to the limitations of the original grayscale features in effectively capturing image characteristics, unlike the deep features utilized by other deep clustering methods. Among the evaluated methods, NCSC achieves the highest clustering results in terms of ACC and ARI. Furthermore, the proposed DISSC algorithm demonstrates exceptional performance in the clustering task by employing nonlocal contrastive self-distillation regularization.

2) Experimental Results on CIFAR-10, STL-10 and Stanford Online Products Datasets: The clustering results on CIFAR-10 and STL-10 datasets illustrate our method significantly outperforms other baselines in terms of ACC, NMI, and ARI. Among the compared methods, FS2CL [56] performs best on STL-10, and SENet [48] excels on CIFAR-10. Due to the high memory and computational costs associated with traditional DSC-Net methods using single-batch training, these approaches are not included in the comparison. It is worth noting that our method demonstrates superior performance compared

to recent scalable deep subspace clustering approaches, such as FS2CL [56] and VDRL-SC [75], demonstrating its effectiveness on large-scale datasets through nonlocal contrastive learning with self-distillation. In contrast to the successful results obtained on the above four datasets, most deep clustering methods struggle to produce satisfactory outcomes on the Stanford Online Products dataset. This can be attributed to the dataset's diverse backgrounds, varying shapes, colors, scales, and view angles. Remarkably, the proposed DISSC method outperforms autoencoder-based deep subspace clustering approaches such as SSC+CAE, LRR+CAE, KSSC+CAE, and NCSC. This highlights the advantages of incorporating nonlocal contrastive learning and self-distillation into subspace clustering.

F. Case Study

To understand the effects of certain choices in the proposed model, several ablations are explored on the MNIST and Fashion-MNIST datasets. Specifically, the model is tested in three settings where: 1) self-distillation (SD) terms are removed from the full model (Model w/o SD); 2) contrastive learning (CL) is removed from the full model (Model w/o NCL) and 3) Full model. In the Model w/o SD, only instance-level contrastive loss is employed using the outputs of the two modules, i.e., $\mathbf{H}_{\mathrm{SE}}$ and $\mathbf{H}_{\mathrm{IN}}$. In the Model w/o CL, the affinities are employed to yield positive or negative data point pairs in the respective modules. Table III reports the results

TABLE III
ABLATION STUDY OF DISSC MODEL IN TERMS OF CLUSTERING PERFORMANCE (IN %) ON MNIST AND FASHION-MNIST DATASETS. BEST CLUSTERING RESULTS ARE HIGHLIGHTED IN BOLD.

Datasets	**MNIST**		
Metrics	**ACC**	**NMI**	**ARI**
Model w/o SD	91.4	88.7	89.1
Model w/o CL	80.1	78.6	77.2
Full model	**98.3**	**94.7**	**95.0**
Datasets	**Fashion-MNIST**		
Metrics	**ACC**	**NMI**	**ARI**
Model w/o SD	70.6	67.3	57.8
Model w/o CL	61.4	61.0	52.6
Full model	**80.5**	**74.5**	**62.0**

of the ablation study. It can be observed that our framework could produce reasonable subspace membership prediction with the help of self-distillation. This stems from the fact that self-distillation encourages the subspace learning module and inductive subspace clustering module to supervise each other, which enhances the model learning in a closed-loop form.

As pointed out in previous literature, data augmentation is an inseparable part of contrastive learning. Consequently, without data augmentation, the loss functions defined in Eqns. (9) and (10) would fail to perform in the way of nonlocal contrastive learning. Instead, it merely works as a role of the graph regularization to preserve the learned subspace information into the intermediate representations. Accordingly, without self-distillation, the SDR in Eqns. (9) and (10) would be omitted. Thus, Model w/o SD cannot propagate the supervision information from subspace learning module to inductive subspace clustering module. Table III reveals that nonlocal contrastive learning makes more contributions to subspace clustering than knowledge self-distillation. In particular, the role of knowledge self-distillation is to guarantee the scalability of our framework without performance sacrifice while, more importantly, nonlocal contrastive learning defines the pretext task for representation learning.

As discussed, our DISSC method addresses the scalability issue that limits most existing deep subspace clustering methods. By employing mini-batch training combined with nonlocal contrastive self-distillation, DISSC effectively handles large-scale datasets and significantly reduces running time. To highlight DISSC's efficiency, Figure 3 compares its running time with that of other methods, including DEC [60], DSC-Net [23], EDESC [73], and FS2CL [56] on the MNIST and Fashion-MNIST datasets. Notably, DSC-Net is less suited for large samples, so comparisons were only performed on a 1,000-sample subset. Even so, DSC-Net's running time exceeds that of other baselines by over ten times. While our method takes slightly more time than some baselines, it maintains impressive clustering performance, demonstrating its effectiveness.

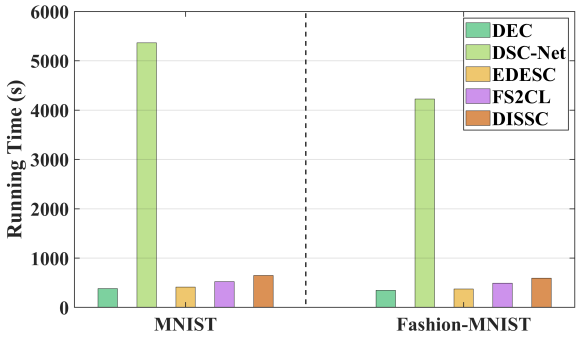
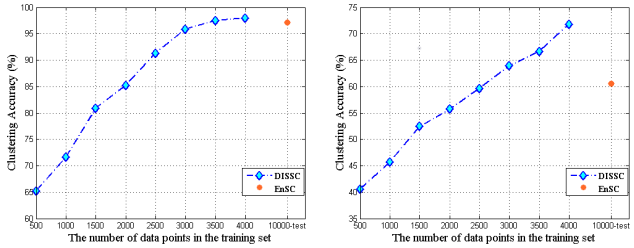


Fig. 3. Running time comparison on MNIST and Fashion-MNIST datasets.

G. Generalization on Out-of-Sample Data

The generalization ability of the proposed DISSC model to out-of-sample data is evaluated using the MNIST and Fashion-MNIST datasets. To assess this, $\{500, 1000, 1500, 2000, 2500, 3000, 3500, 4000\}$ data points are randomly selected from each category in MNIST-train and Fashion-MNIST-train for training our model. Subsequently, data points from MNIST-test and Fashion-MNIST-test are utilized to evaluate the generalization performance. Figure 4 illustrates the results in terms of clustering accuracy on MNIST-test and Fashion-MNIST-test datasets. As depicted, as the number of training data points increases, our framework demonstrates the ability to predict more accurate cluster labels for unseen data. This ultimately leads to superior clustering performance compared to EnSC [32], which is directly optimized using the MNIST-test and Fashion-MNIST-test datasets. In Fig. 4, the clustering result by utilizing the EnSC algorithm is marked as "10000-test", denoting the utilization of 10000 data points in the training set. Notably, the proposed DISSC algorithm surpasses EnSC in accurately predicting cluster labels

for data points in the testing set, even with a reduced number of training data points—specifically, 3500 and 3000 training data points for the MNIST and Fashion-MNIST datasets, respectively. It is important to highlight that the autoencoder-based deep subspace clustering networks operate based on transductive clustering principles, and therefore struggle to directly determine the cluster membership of unseen data.



(a) MNIST-test (b) Fashion-MNIST-test

Fig. 4. Generalization performance of our model on (a) MNIST-test and (b) Fashion-MNIST-test with different number of data points for training.

H. Parameter Analysis

1) Analysis on batch size: Since the proposed DISSC framework is trained in a batch-by-batch manner, the effect of batch size on clustering performance is investigated in this subsection. Figure 5 shows that larger batch sizes have a significant advantage over the smaller ones. This is the reason why larger batch sizes would contain more data, which not only enables our framework to learn sufficient and meaningful subspace information by capturing self-expression property in a big batch of data but also provides more positive and negative pairs to facilitate the nonlocal contrastive learning for desirable representations.

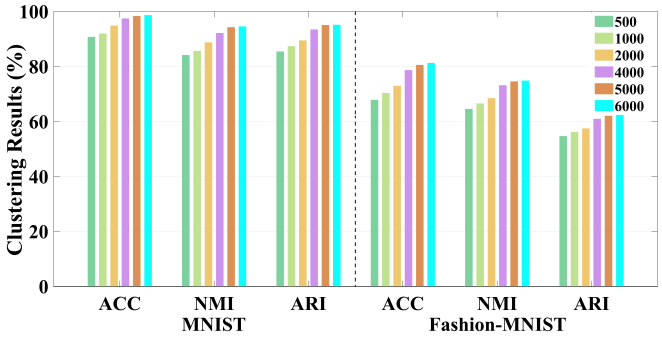
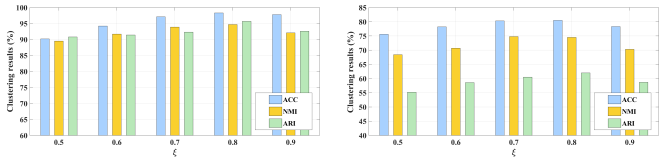


Fig. 5. Clustering performance of the proposed DISSC model on MNIST-test and Fashion-MNIST-test with different batch sizes and same training epochs.

2) Analysis on threshold ξ: As mentioned above, the affinities $\boldsymbol{A}_{\mathrm{SE}}$ and $\boldsymbol{A}_{\mathrm{IN}}$ derived from the two modules have the advantage in positive and negative supervision mining, respectively. In this experiment, the proposed algorithm is repeated on MNIST and Fashion-MNIST datasets with different settings of ξ. Figure 6 shows the clustering results in terms of ACC, NMI, and ARI vs. ξ on MNIST and Fashion-MNIST datasets in the set of $\{0.5, 0.6, 0.7, 0.8, 0.9\}$. From Fig. 6, it can be seen that the better performance prefers to a larger number of threshold. In the experiments, ξ is empirically set as 0.8. Note that a specific threshold may improve the clustering results.

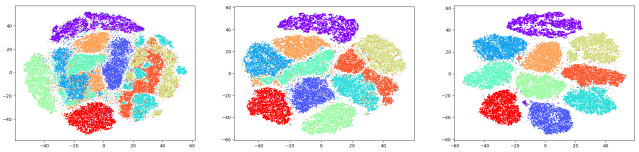


(a) MNIST (b) Fashion-MNIST

Fig. 6. The clustering results in terms of ACC, NMI, and ARI v.s. ξ on MNIST and Fashion-MNIST datasets.

I. Qualitative Analysis

To further understand the mechanism of self-distillation, the t-SNE [76] is utilized to visualize intermediate representations captured at different phases by plotting them on a 2D map. As shown in Fig. 7, the intermediate representations are all mixed and most data points are assigned to a few subspaces at the very beginning. Although pre-training can equip our model with preliminary discriminative ability, it is still far from enough to generate reliable soft labels to infer subspace membership. By jointly optimizing the whole network with self-distillation, the inductive subspace clustering module could predict reasonable subspace assignment and the subspace learning module could promote the intermediate presentations to be subspace-preserving, learning discriminative representations and desirable cluster assignments at the same time.



(a) without training (NMI=5.74%) (b) with pre-training (NMI=68.3%) (c) with fine-tuning (NMI=94.7%)

Fig. 7. The evolution of intermediate representations throughout the whole training process on MNIST. Points with the same color indicate data predicted to lie in the same subspace.

V. Conclusion

This paper delved deeply into the existing work on deep subspace clustering in large-scale data and proposed a novel DISSC algorithm for deep subspace clustering by designing a nonlocal contrastive prediction task, where the subspace membership of data can be directly inferred without seeking help from spectral clustering. By integrating self-distillation with deep subspace clustering, the proposed framework can be trained in a batch-by-batch manner to be applied to large-scale online scenarios without sacrificing the clustering performance. The extensive experiments on large-scale benchmark datasets demonstrate that the proposed DISSC framework significantly outperforms the state-of-the-art subspace clustering methods.

The success of contrastive self-distillation in deep subspace clustering presents a promising solution to scalability challenges in large-scale datasets. However, the affinities between data samples within a mini-batch are significantly influenced by complementary knowledge drawn from multiple perspectives. The presence of highly homogeneous knowledge complicates the effective enhancement of affinities through the self-distillation mechanism. This issue becomes particularly pronounced in fine-grained image clustering tasks, where capturing local features and refining affinities will be a key focus of our future research in subspace clustering.

ACKNOWLEDGMENTS

Thanks for the resources provided by the School of Engineering, Computer and Mathematical Sciences at Auckland University of Technology New Zealand during the visiting time.

REFERENCES

[1] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, 2013.

[2] L. Xing, B. Chen, J. Wang, S. Du, and J. Cao, "Robust high-order manifold constrained low rank representation for subspace clustering," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 31, no. 2, pp. 533–545, 2020.

[3] Y. Jia, G. Lu, H. Liu, and J. Hou, "Semi-supervised subspace clustering via tensor low-rank representation," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 33, no. 7, pp. 3455–3461, 2023.

[4] Z. Peng, Y. Jia, H. Liu, J. Hou, and Q. Zhang, "Maximum entropy subspace clustering network," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 32, no. 4, pp. 2199–2210, 2022.

[5] Z. Zhou, C. Ding, J. Li, E. Mohammadi, G. Liu, Y. Yang, and Q. J. Wu, "Sequential order-aware coding-based robust subspace clustering for human action recognition in untrimmed videos," *IEEE Trans. Image Process.*, vol. 32, pp. 13–28, 2022.

[6] J. Lei, X. Li, B. Peng, L. Fang, N. Ling, and Q. Huang, "Deep spatial-spectral subspace clustering for hyperspectral image," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 31, no. 7, pp. 2686–2697, 2020.

[7] B. Wang, Y. Hu, J. Gao, Y. Sun, F. Ju, and B. Yin, "Learning adaptive neighborhood graph on grassmann manifolds for video/image-set subspace clustering," *IEEE Trans. Multimedia*, vol. 23, pp. 216–227, 2020.

[8] Y.-P. Zhao, X. Dai, Z. Wang, and X. Li, "Subspace clustering via adaptive non-negative representation learning and its application to image segmentation," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 33, no. 8, pp. 4177–4189, 2023.

[9] X. Peng, L. Zhang, and Z. Yi, "Scalable sparse subspace clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 430–437.

[10] W. Zhu and B. Peng, "Sparse and low-rank regularized deep subspace clustering," *Knowl.-Based Syst.*, vol. 204, p. 106199, 2020.

[11] R. Vidal and P. Favaro, "Low rank subspace clustering (LRSC)," *Pattern Recog. Lett.*, vol. 43, pp. 47–61, 2014.

[12] H. Zhang, S. Li, J. Qiu, Y. Tang, J. Wen, Z. Zha, and B. Wen, "Efficient and effective nonconvex low-rank subspace clustering via svt-free operators," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 33, no. 12, pp. 7515–7529, 2023.

[13] J. Zhou, C. Huang, C. Gao, Y. Wang, W. Pedrycz, and G. Yuan, "Reweighted subspace clustering guided by local and global structure preservation," *IEEE Trans. Cybern.*, vol. 55, no. 3, pp. 1436–1449, 2025.

[14] Y. Xu, S. Chen, J. Li, and J. Yang, "Metric learning-based subspace clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 6, pp. 10 491–10 503, 2025.

[15] L. Bai and J. Liang, "Sparse subspace clustering with entropy-norm," in *Proc. Int. Conf. Mach. Learn.* PMLR, 2020, pp. 561–568.

[16] S. Wang, Y. Chen, Z. Lin, Y. Cen, and Q. Cao, "Robustness meets low-rankness: Unified entropy and tensor learning for multi-view subspace clustering," *IEEE Trans. Circuit Syst. Video Technol.*, 2023.

[17] M. Sun, S. Wang, P. Zhang, X. Liu, X. Guo, S. Zhou, and E. Zhu, "Projective multiple kernel subspace clustering," *IEEE Trans. Multimedia*, vol. 24, pp. 2567–2579, 2022.

[18] X. Zhang, B. Chen, H. Sun, Z. Liu, Z. Ren, and Y. Li, "Robust low-rank kernel subspace clustering based on the schatten p-norm and correntropy," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 12, pp. 2426–2437, 2020.

[19] C. Peng, Q. Zhang, Z. Kang, C. Chen, and Q. Cheng, "Kernel two-dimensional ridge regression for subspace clustering," *Pattern Recog.*, vol. 113, p. 107749, 2021.

[20] X. Zhang, S. Zhao, J. Wang, L. Guo, X. Wang, and H. Sun, "Purity-preserving kernel tensor low-rank learning for robust subspace clustering," *IEEE Trans. Circuit Syst. Video Technol.*, 2023.

[21] X. Peng, S. Xiao, J. Feng, W.-Y. Yau, and Z. Yi, "Deep subspace clustering with sparsity prior." in *Proc. Int. Joint Conf. on Artif. Intell.*, 2016, pp. 1925–1931.

[22] X. Peng, J. Feng, J. T. Zhou, Y. Lei, and S. Yan, "Deep subspace clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 12, pp. 5509–5521, 2020.

[23] P. Ji, T. Zhang, H. Li, M. Salzmann, and I. Reid, "Deep subspace clustering networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017.

[24] Y. Chen, W. Wu, L. Ou-Yang, R. Wang, and S. Kwong, "GRESS: Grouping belief-based deep contrastive subspace clustering," *IEEE Trans. Cybern.*, vol. 55, no. 1, pp. 148–160, 2025.

[25] P. Zhou, Y. Hou, and J. Feng, "Deep adversarial subspace clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1596–1604.

[26] J. Zhang, C. Li, C. You, X. Qi, H. Zhang, J. Guo, and Z. Lin, "Self-supervised convolutional subspace clustering network," in *Proc. 32th IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5468–5477.

[27] Z. Kang, X. Xie, B. Li, and E. Pan, "CDC: A simple framework for complex data clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 7, pp. 13 177–13 188, 2025.

[28] Z. Wang, J. Zhao, C. Lu, H. Huang, F. Yang, L. Li, and Y. Guo, "Learning to detect head movement in unconstrained remote gaze estimation in the wild," in *IEEE Winter Applications Comput. Vis. Conf.*, 2020, pp. 3432–3441.

[29] B. Huang, J. Li, J. Chen, G. Wang, J. Zhao, and T. Xu, "Anti-UAV410: A thermal infrared benchmark and customized scheme for tracking drones in the wild," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 5, pp. 2852–2865, 2024.

[30] J. Zhao, J. Li, H. Liu, S. Yan, and J. Feng, "Fine-grained multi-human parsing," *Int. J. Comput. Vis.*, vol. 128, no. 5, pp. 2185–2203, 2020.

[31] K. Xu, L. Chen, and S. Wang, "Towards robust nonlinear subspace clustering: A kernel learning approach," *IEEE Trans. Artif. Intell.*, pp. 1–13, 2025.

[32] C. You, C.-G. Li, D. P. Robinson, and R. Vidal, "Oracle based active set algorithm for scalable elastic net subspace clustering," in *Proc. 29th IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3928–3937.

[33] C. You, D. Robinson, and R. Vidal, "Scalable sparse subspace clustering by orthogonal matching pursuit," in *Proc. 29th IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3918–3927.

[34] Z. Kang, Z. Lin, X. Zhu, and W. Xu, "Structured graph learning for scalable subspace clustering: From single view to multiview," *IEEE Trans. Cybern.*, vol. 52, no. 9, pp. 8976–8986, 2022.

[35] L. Zhou, B. Xiao, X. Liu, J. Zhou, E. R. Hancock *et al.*, "Latent distribution preserving deep subspace clustering," in *Proc. Int. Joint Conf. Artif. Intell.*, 2019, pp. 4440–4446.

[36] S. Wu and W.-S. Zheng, "Semisupervised feature learning by deep entropy-sparsity subspace clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 2, pp. 774–788, 2022.

[37] S. Huang, H. Zhang, and A. Pižurica, "Subspace clustering for hyperspectral images via dictionary learning with adaptive regularization," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–17, 2022.

[38] Z. Kang, X. Lu, J. Liang, K. Bai, and Z. Xu, "Relation-guided representation learning," *Neural Netw.*, vol. 131, pp. 93–102, 2020.

[39] J. M. Jose Valanarasu and V. M. Patel, "Overcomplete deep subspace clustering networks," in *Proc. IEEE Wint. Appl. Comput. Vis.*, 2021, pp. 746–755.

[40] Z. Dang, C. Deng, X. Yang, and H. Huang, "Multi-scale fusion subspace clustering using similarity constraint," in *Proc. 33th IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6657–6666.

[41] M. Abavisani, A. Naghizadeh, D. Metaxas, and V. Patel, "Deep subspace clustering with data augmentation," in *Proc. 34th Adv. Neural Inf. Process. Syst.*, 2020, pp. 10 360–10 370.

[42] K. Li, H. Liu, Y. Zhang, K. Li, and Y. Fu, "Self-guided deep multiview subspace clustering via consensus affinity regularization," *IEEE Trans. Cybern.*, vol. 52, no. 12, pp. 12 734–12 744, 2022.

[43] G. He, W. Jiang, R. Peng, M. Yin, and M. Han, "Soft subspace based ensemble clustering for multivariate time series data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 10, pp. 7761–7774, 2023.
[44] Z. Yu, Z. Zhang, W. Cao, C. Liu, C. L. P. Chen, and H.-S. Wong, "GAN-based enhanced deep subspace clustering networks," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 7, pp. 3267–3281, 2022.
[45] J. Lv, Z. Kang, X. Lu, and Z. Xu, "Pseudo-supervised deep subspace clustering," *IEEE Trans. Image Process.*, vol. 30, pp. 5252–5263, 2021.
[46] P. Zhang, W. Zhu, and W. Q. Yan, "Multi-level structural contrastive subspace clustering network," *IEEE Signal Processing Letters*, vol. 32, pp. 3092–3096, 2025.
[47] T. Zhang, P. Ji, M. Harandi, W. Huang, and H. Li, "Neural collaborative subspace clustering," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 7384–7393.
[48] S. Zhang, C. You, R. Vidal, and C.-G. Li, "Learning a self-expressive network for subspace clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 12 388–12 398.
[49] T. Zhang, P. Ji, M. Harandi, R. Hartley, and I. Reid, "Scalable deep k-subspace clustering," in *Proc. Asian Conf. Comput. Vis.*, 2018, pp. 466–481.
[50] J. Busch, E. Faerman, M. Schubert, and T. Seidl, "Learning self-expression metrics for scalable and inductive subspace clustering," *NeurIPS 2020 Workshop*, 2020.
[51] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *Int. J. Comput. Vis.*, no. 129, pp. 1789–1819, 2021.
[52] Y. Li, P. Hu, Z. Liu, D. Peng, J. T. Zhou, and X. Peng, "Contrastive clustering," in *Proc. 35th AAAI Conf. on Artif. Intell.*, 2021, pp. 8547–8555.
[53] E. Pan and Z. Kang, "Multi-view contrastive graph clustering," *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, pp. 2148–2159, 2021.
[54] Y. Li, M. Yang, D. Peng, T. Li, J. Huang, and X. Peng, "Twin contrastive learning for online clustering," *Int. J. Comput. Vis.*, vol. 130, no. 9, pp. 2205–2221, 2022.
[55] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.
[56] Q. Wang, X. Ye, and N. Wang, "Learning low-rank representation approximation for few-shot deep subspace clustering," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 34, no. 11, pp. 10 590–10 603, 2024.
[57] S. G. Müller and F. Hutter, "Trivialaugment: Tuning-free yet state-of-the-art data augmentation," *arXiv preprint arXiv:2103.10158*, 2021.
[58] H. Liu, Z. Wu, X. Li, D. Cai, and T. S. Huang, "Constrained nonnegative matrix factorization for image representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1299–1311, 2011.
[59] J. Munkres, "Algorithms for the assignment and transportation problems," *SIAM J.*, vol. 5, no. 1, pp. 32–38, 1957.
[60] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *Proc. 33th Int. Conf. Mach. Learn.*, 2016, pp. 478–487.
[61] X. Guo, L. Gao, X. Liu, and J. Yin, "Improved deep embedded clustering with local structure preservation," in *Proc. Int. Joint Conf. Artif. Intell.*, vol. 17, 2017, pp. 1753–1759.
[62] J. Yang, D. Parikh, and D. Batra, "Joint unsupervised learning of deep representations and image clusters," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 5147–5156.
[63] J. Chang, L. Wang, G. Meng, S. Xiang, and C. Pan, "Deep adaptive image clustering," in *Proc. 16th IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5879–5887.
[64] B. Yang, X. Fu, N. D. Sidiropoulos, and M. Hong, "Towards k-means-friendly spaces: Simultaneous deep learning and clustering," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 3861–3870.
[65] K. Ghasedi Dizaji, A. Herandi, C. Deng, W. Cai, and H. Huang, "Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization," in *Proc. 16th IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5736–5745.
[66] S. Mukherjee, H. Asnani, E. Lin, and S. Kannan, "ClusterGAN: Latent space clustering in generative adversarial networks," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 4610–4617.
[67] J. Huang, S. Gong, and X. Zhu, "Deep semantic clustering by partition confidence maximisation," in *Proc. IEEE / CVF Comput. Vis. Pattern Recognit. Conf.*, 2020, pp. 8846–8855.
[68] J. Cai, S. Wang, C. Xu, and W. Guo, "Unsupervised deep clustering via contractive feature representation and focal loss," *Pattern Recognit.*, p. 108386, 2022.
[69] L. Yang, W. Fan, and N. Bouguila, "Clustering analysis via deep generative models with mixture models," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 1, pp. 340–350, 2022.
[70] T. Wang, X. Zhang, L. Lan, and Z. Luo, "Local-to-global deep clustering on approximate uniform manifold," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 5, pp. 5035–5046, 2023.
[71] J. Cai, Y. Zhang, S. Wang, J. Fan, and W. Guo, "Wasserstein embedding learning for deep clustering: A generative approach," *IEEE Trans. Multimedia*, vol. 26, pp. 7567–7580, 2024.
[72] C. You, C. Li, D. P. Robinson, and R. Vidal, "Scalable exemplar-based subspace clustering on class-imbalanced data," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 67–83.
[73] J. Cai, J. Fan, W. Guo, S. Wang, Y. Zhang, and Z. Zhang, "Efficient deep embedded subspace clustering," in *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 21–30.
[74] L. Wei, Z. Chen, J. Yin, C. Zhu, R. Zhou, and J. Liu, "Adaptive graph convolutional subspace clustering," in *Proc. IEEE / CVF Comput. Vis. Pattern Recognit. Conf.*, 2023, pp. 6262–6271.
[75] Y. Li, S. Wang, C. Li, Y. Yuan, and G. Wang, "Towards very deep representation learning for subspace clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 7, pp. 3568–3579, 2024.
[76] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, 2008.

Wenjie Zhu (Member, IEEE) received the Master's degree from Xidian University, Xi'an, China, in 2013 and the Ph.D. degree from Northeastern University, Shenyang, China, in 2019. He visited the School of Engineering, Computer & Mathematical Sciences at Auckland University of Technology (AUT) in New Zealand from 2023 to 2024. Currently, he is a lecturer in the College of Information Engineering in China Jiliang University, Hangzhou, China. His current research interests include computer vision and machine learning.

Bo Peng received the Bachelor's degrees from China Jiliang University in Hangzhou, China, and Auckland University of Technology (AUT) in Auckland, New Zealand in 2020. He received the Master's degree from The University of Queensland in 2023. His research interests include pattern recognition and machine learning.

Wei Qi Yan (Senior Member, IEEE) is with AUT computer science, his expertise covers intelligent surveillance, deep learning, robotics, computer vision, and multimedia computing. Dr. Yan has served as an Associate Editor of ACM Transactions on Multimedia Computing, Communications and Applications (TOMM), an Associate Editor of Frontiers in Neuroscience, an Associate Editor of Springer Nature Computer Science, the Editor-in-Chief (EiC) of the International Journal of Digital Crime and Forensics (IJDCF), and he has worked as an exchange computer scientist between the Royal Society Te Apārangi (RSNZ) and the Chinese Academy of Sciences (CAS) in China. He is a guest (adjunct) professor at the Chinese Academy of Sciences and has been a visiting professor at the University of Auckland in New Zealand and the National University of Singapore. In 2022, Dr. Yan was recognised as one of the world's top 2% cited scientists by Stanford University, USA. He currently holds the position of Chair of ACM Multimedia Chapter of New Zealand and a member of the ACM, a senior member of the IEEE and a TC member of the IEEE.