

# Hierarchical Graph Learning for Spectral Clustering: From Single View to Multiview

## Abstract

Spectral clustering is widely recognized for its effectiveness in segmenting data, particularly in identifying complex cluster structures within high-dimensional spaces. However, its performance in both single-view and multi-view contexts heavily relies on the construction of the similarity graph. Traditional non-hierarchical methods often fail to capture multi-level structural information, and the linear nature of the graph Laplacian matrix limits spectral clustering's ability to manage complex nonlinear data structures. To overcome these challenges, we propose a novel hierarchical graph structure, the constrained 1-nearest neighbor (CNN) graph, which simultaneously captures both local and global nonlinear structures. Building on this, we introduce two enhanced spectral clustering algorithms: Spectral Clustering based on CNN graph (SC-CNN), a single-view approach that utilizes the CNN graph for dimensionality reduction, followed by the reconstruction of a more robust similarity graph; and Multi-View Spectral Clustering based on CNN graph (MVSC-CNN), a multi-view extension that incorporates multi-level structural information from each view with ensemble clustering. Extensive experiments on real-world datasets demonstrate that both SC-CNN and MVSC-CNN consistently outperform state-of-the-art clustering methods, offering superior accuracy and robustness.

## Introduction

Spectral clustering (SC) is a powerful technique for data segmentation, particularly for identifying complex structures in high-dimensional spaces. Unlike traditional methods based on Euclidean distances, it leverages the eigenvalues and eigenvectors of the graph Laplacian matrix to partition data, capturing non-linear relationships effectively. However, its performance is highly dependent on the choice of graph structure (Ng et al., 2002; von Luxburg, 2007), making it sensitive to noise and the representation of the data's intrinsic geometry. Additionally, the linear nature of the graph Laplacian matrix limits its ability to fully capture more complex data structures, such as multi-modal clusters, highlighting the need for alternative methods to enhance clustering performance (Jianbo Shi & Malik, 2000).

In SC, the construction of the affinity (or similarity) graph is crucial for determining clustering outcomes. Common approaches include the  $k$ -nearest neighbor ( $k$ -NN) graph, which connects each node to its  $k$  nearest neighbors. How-

ever,  $k$ -NN can mistakenly link points from two closely located clusters, leading to potential misclusterings (von Luxburg, 2007). The mutual  $k$ -nearest neighbor (Mk-NN) graph, which only connects mutual nearest neighbors, creates a sparser graph, but can result in an excessive number of disconnected components, reducing intra-cluster cohesion and diminishing clustering effectiveness (Tan et al., 2020). The  $\epsilon$ -neighborhood graph connects points within a set distance  $\epsilon$ , capturing dense regions but potentially missing connections in sparser areas (İnkaya, 2015). The fully connected graph links all points with edges weighted by their similarity, offering a comprehensive view of data relationships but at the cost of increased computational complexity and sensitivity to the similarity scale parameter (Zelnik-Manor & Pietro, 2004). The bipartite graph, which divides nodes into two sets, may overlook intra-set similarities, potentially leading to suboptimal clustering performance (H. Zhang et al., 2023). Recently, researchers have also proposed various other graph structures to address specific challenges in SC (Alshammari et al., 2021; Y. Cai et al., 2022; İnkaya et al., 2015; X. Zhang et al., 2011). However, like the aforementioned methods, these structures primarily capture local structures or single-level information, lacking the capability to represent both local and global structures simultaneously or to effectively capture multi-granularity information as hierarchical graphs do.

The original SC directly uses the graph Laplacian matrix to capture cluster manifolds, but its linear nature limits its ability to handle complex nonlinear data structures, such as multimodal distributions. To address this, recent approaches have integrated more cluster-friendly low-dimensional neural network representations with SC, resulting in more robust performance (Affeldt et al., 2020; Law et al., 2017; X. Li, Wei, et al., 2022; Shaham et al., 2018a; X. Yang et al., 2019; Zhao & Li, 2023a, 2023b). For example, (Shaham et al., 2018a) introduced SpectralNet, which employs neural network to embed data into the eigenspace of the Laplacian matrix, improving clustering performance on nonlinear data. (Affeldt et al., 2020) proposed an ensemble framework that integrates SC with deep autoencoders to address robustness issues in deep clustering methods. Their model generates diverse embeddings and refines clustering through a sparse affinity matrix, aiming to capture complex data relationships. (S. Huang et al., 2019) introduced MultiSpectralNet (MvSN)

to improve spectral multi-view clustering by correcting misleading information from individual views. This approach maps data to fusion eigenvectors, enabling better handling of nonlinear data structures across multiple views. However, all these methods rely on the cluster labels from non-hierarchical algorithms to guide the generation of low-dimensional embeddings, which cannot provide more discriminative multi-level, multi-granularity information, resulting in suboptimal clustering performance on more complex data.

Additionally, as an extension of SC, multi-view SC (D. Huang et al., 2020, 2023; Wen et al., 2020, 2021) is also a key technique that merges similarity graphs from multiple data views into a unified graph, capturing both shared and unique structures to enhance clustering accuracy through effective graph fusion. For instance, (Kang, Shi, et al., 2020) combined view-specific graphs into a consensus graph, (Tang et al., 2023) used unified non-negative embeddings for a coherent graph, and (Ren et al., 2024) dynamically fused latent graphs to capture cross-view topological information. However, most existing methods focus on fusing similarity graph structures from different views but still rely on single-granularity structural information. This reliance limits their ability to effectively capture multi-level information, resulting in the loss of critical discriminative details and leading to suboptimal performance on more complex datasets.

This paper introduces a novel hierarchical similarity graph to address three critical issues in current spectral clustering (SC) methods: the reliance on non-hierarchical graphs that overlook global and multi-granularity structural information; the inability of existing SC methods to capture complex nonlinear structures due to the linear nature of the graph Laplacian derived from raw data; and the neglect of discriminative structural information across different granularities in current multi-view SC methods. Specifically, we propose a new hierarchical similarity graph, the Constrained 1-Nearest Neighbor (CNN) graph, and introduce two advanced SC algorithms based on it: spectral clustering using the CNN graph (SC-CNN) and multi-view spectral clustering using the CNN graph (MVSC-CNN). SC-CNN leverages the connected components of the CNN graph at a specific granularity to guide the learning of a more cluster-friendly low-dimensional embedding, which is then used to reconstruct the CNN graph, serving as input for SC and significantly enhancing its generalization capabilities. MVSC-CNN constructs CNN graphs for each view and combines the connected components across granularities using ensemble clustering. This method derives a latent representation that integrates multi-granularity, local, and global structural information, further improving SC performance on multi-view data. The key contributions of this paper are summarized as follows:

- We propose a novel hierarchical similarity graph, the Constrained 1-Nearest Neighbor (CNN) graph, capable of capturing more complex local and global structures, such as modality heterogeneity in data distribution.

- We introduce an enhanced single-view spectral clustering method, SC-CNN, which leverages cluster-friendly embeddings learned from the CNN graph at specific granularities, along with the superior similarity representation provided by the CNN graph.
- We develop an enhanced multi-view spectral clustering method, MVSC-CNN, which integrates multi-granularity structural information by fusing CNN graphs across views.
- Extensive experiments on multiple single-view and multi-view real-world datasets demonstrate the superiority of our enhanced spectral clustering methods.

## Related Work

### Spectral Clustering

Spectral clustering is a technique that leverages the eigenstructure of a similarity matrix to partition data into clusters (Ng et al., 2002; von Luxburg, 2007). Given a dataset  $X = \{x_1, x_2, \dots, x_n\}$ , spectral clustering begins by constructing a similarity graph  $G = (V, E)$ , where  $V$  represents the data points and  $E$  the weighted edges between them, typically defined by a similarity function  $s(x_i, x_j)$ . The next step involves forming the graph's Laplacian matrix  $L = D - W$ , where  $W$  is the weighted adjacency matrix and  $D$  is the diagonal degree matrix with  $D_{ii} = \sum_j W_{ij}$ . The normalized Laplacian can be defined as:

$$L_{\text{sym}} = I - D^{-\frac{1}{2}}WD^{-\frac{1}{2}} \quad (1)$$

Spectral clustering then solves an eigenvalue problem, finding the first  $NC$  eigenvectors of  $L_{\text{sym}}$ , forming the matrix  $U$ . These eigenvectors are treated as new representations of the data points. Finally,  $k$ -means or another clustering algorithm is applied to the rows of  $U$  to obtain the final clusters.

### Deep Clustering

Deep clustering is a technique where cluster labels are treated as ground-truth labels, combined with neural networks to guide the learning of cluster-friendly representations and enhance clustering performance. Since cluster labels may contain errors, many methods focus on refining them. For example, (X. Zhang et al., 2021) improved cluster label quality by assessing the similarity of labels across training iterations using clustering consensus, while (S. Park et al., 2021) removed incorrect labels through label smoothing. (Lu et al., 2022) selected cluster labels near cluster centers as reliable, (S. Wang et al., 2022) used cluster-soft labels to reduce noise, and (Mahon & Lukasiewicz, 2021) employed an ensemble strategy to fuse cluster labels from multiple algorithms to improve accuracy and reliability in deep clustering models. However, these methods do not consider using multi-granularity cluster labels, limiting their ability to extract more discriminative representations.

## Multi-view Spectral Clustering

Multi-view spectral clustering is a technique that integrates similarity graphs from multiple data views into a unified graph, capturing complementary information across views to enhance clustering performance. (Kang, Shi, et al., 2020) constructs separate similarity graphs for each view and then fuses them into a consensus graph that balances the contributions of each view, effectively capturing both local and global structures. (Tang et al., 2023) proposes a unified framework that uses specific and unified non-negative embeddings to integrate graph structures from different views into a coherent graph, leading to more accurate clustering results. (Ren et al., 2024) introduces a dynamic fusion mechanism that constructs latent graphs for each view, which are then combined into a fusion graph that adapts to the topological information of the data, thereby enhancing clustering performance.

## The Proposed Method

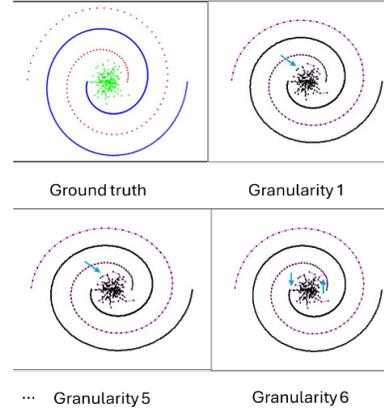
### Constrained 1-Nearest Neighbor (CNN) graph

Inspired by previous work on constrained hierarchical clustering (J. Yang & Lin, 2024), which measures the size relationship between each (sub)-cluster and its 1-nearest cluster to guide constrained hierarchical merging, we develop the Constrained 1-Nearest Neighbor (CNN) graph. The CNN graph captures both local and global structures across different granularities, effectively increasing similarity within clusters while reducing similarity between different clusters.

Given a dataset  $X = \{x_1, x_2, \dots, x_n\}$ , where each sample  $x_i$  corresponds to a node  $u_i$ , suppose the 1-nearest neighbor of  $u_i$  is  $u_{i_N}$ . The similarity between nodes  $u_i$  and  $u_{i_N}$  is defined as:

$$s(u_i, u_{i_N}) = \begin{cases} e^{-m_{u_i} \cdot m_{u_{i_N}} \cdot d(i, i_N)^2}, & \text{if } m_{u_i} \leq m_{u_{i_N}} \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

Here,  $d(i, i_N)$  represents the distance between  $u_i$  and  $u_{i_N}$ , and  $m_{u_i}$  denotes the number of nodes in the sub-graph containing  $u_i$ . Initially, both  $m_{u_i}$  and  $m_{u_{i_N}}$  are equal to 1, as each sub-graph starts with only a single node. This method of constructing the similarity graph proceeds hierarchically. From the second iteration onward,  $u_i$  and  $u_{i_N}$  represent the closest nodes from two neighboring sub-graphs. The process continues by repeatedly applying Eq. (2) until the similarity graph has only a single connected component, at which point the construction of the similarity graph is complete. As shown in Fig. 1, we construct the CNN graph on a 2D dataset with multimodal and varying density distributions. At granularity 1, most connections are established, primarily capturing the local structure of the data. By granularity 5, each sub-graph corresponds to a ground-truth cluster. At granularity 6, all connections are completed, capturing the global structure of the data at this level. This hierarchical



**Figure 1.** Hierarchically construct the CNN graph on a 2D dataset.

method effectively captures both local and global structures, handling multimodal distributions and enhancing spectral clustering performance. Moreover, it eliminates the need for specifying parameters, such as the number of neighbors in k-NN graphs, offering significant convenience.

### Spectral Clustering using the CNN graph (SC-CNN)

In this section, we propose SC-CNN, which enhances single-view spectral clustering through the CNN graph in two key ways: by learning a low-dimensional embedding from the CNN graph of the original dataset and by reconstructing the CNN graph using the learned embedding. The first approach leverages the discriminative data structure information captured at a specific granularity within the CNN graph to produce a more cluster-friendly representation. The second approach further enhances the clusterability of the similarity matrix by increasing intra-cluster similarity and reducing inter-cluster similarity.

#### Learning low-dimensional embedding using the CNN graph

Given the original dataset  $X$  with dimensionality  $d$ , we first construct the CNN graph using Eq. (2). Let  $C_k$  represent the connected components of the CNN graph at a specific granularity  $k$ , which are considered ground-truth labels for training a shallow MLP with a single hidden layer. The forward propagation from input  $X$  to the output layer is represented as:

$$\hat{Y} = \text{softmax}(W_2 \cdot \tanh(W_1 \cdot X + b_1) + b_2) \quad (3)$$

Here,  $W_1 \in \mathbb{R}^{d_h \times d}$  and  $W_2 \in \mathbb{R}^{C_k \times d_h}$  are the weight matrices for the hidden and output layers, respectively, where  $d_h$  is the number of neurons in the hidden layer (with  $d_h \ll d$ ), and  $C_k$  is the number of connected components in the CNN graph at granularity  $k$ .  $b_1$  and  $b_2$  are the biases, with hyperbolic tangent ( $\tanh$ ) used as the activation function in the hidden layer and softmax in the output layer. Once the MLP is trained using a gradient descent algorithm, the learned low-dimensional representation of  $X$  is given by:

$$Z = W_1 \cdot X + b_1 \quad (4)$$

Here,  $Z \in \mathbb{R}^{d_h \times n}$  represents the newly learned  $d_h$ -dimensional embedding for the dataset  $X$ .

The reason for using an MLP with only a single hidden layer is its superior interpretability and faster training efficiency compared to more complex network structures. For instance, the newly learned representation  $Z$  in Eq. (4) is merely a linear transformation of the original dataset  $X$ , which greatly preserves interpretability. Additionally, in our implementation, we consistently fixed  $d_h$  to 100, eliminating the need for further tuning. Finally, for granularity  $k$ , we typically select the level that produces a number of connected components greater than the ground-truth number of clusters (NC). This choice ensures that the labels contain more discriminative information, which is beneficial for learning a cluster-friendly embedding (Chen et al., 2019).

### Reconstructing the CNN graph based on the learned embedding

As a graph-based clustering algorithm, spectral clustering requires a similarity matrix as input. Therefore, after learning the cluster-friendly embedding, we reconstruct the full CNN graph using this embedding as the similarity matrix input for spectral clustering. The pseudocode for the proposed SC-CNN is shown in Algorithm 1.

### Multi-view Spectral Clustering using the CNN graph (MVSC-CNN)

In this section, we present an enhanced multi-view spectral clustering method using the CNN graph (MVSC-CNN), which leverages the CNN graph to capture multi-scale structures across multiple views. This method addresses the limitations of traditional multi-view spectral clustering, which only considers single-level structural information in graphs, by integrating data from all views at different levels of granularity, resulting in a more robust similarity matrix for spectral clustering.

### Construction and aggregation of CNN graphs across multiple views

For a given multi-view dataset  $X = \{X_{v_i}\}$ , where each  $X_{v_i}$  represents the data from view  $v_i$ , we construct a CNN graph for each view. In each view  $v_i$ , the CNN graph is built based on the data  $X_{v_i}$ , and connected components are extracted at all granularities  $k$ , ranging from the finest to the coarsest granularity. To clarify the notation,  $C_{v_i}^k$  represents the set of connected components at granularity  $k$  for the CNN graph of view  $v_i$ . Here,  $v_i$  identifies the specific view, while  $k$  denotes the granularity level. The parameter  $k_{\max}$  refers to the maximum granularity level considered, where the graph captures the broadest connections, typically resulting in the entire dataset being connected as a single component. The connected components across all views and granularities are then aggregated into a comprehensive set, represented as:

$$\{\{C_{v_1}^1, C_{v_1}^2, \dots, C_{v_1}^{k_{\max}}\}, \{C_{v_2}^1, C_{v_2}^2, \dots, C_{v_2}^{k_{\max}}\}, \dots, \{C_{v_{\max}}^1, C_{v_{\max}}^2, \dots, C_{v_{\max}}^{k_{\max}}\}\} \quad (5)$$

This aggregation allows the method to capture multi-level structure information, where each connected component acts as a cluster label contributing to the similarity between data points across different views.

### Construction of multi-view consensus similarity matrix

The final step involves constructing the similarity matrix  $S$ , which serves as input for the spectral clustering process. Inspired by techniques from ensemble clustering (D. Huang et al., 2021), the similarity between any two samples  $x$  and  $y$  is computed using a consensus similarity calculation method that integrates information from all connected components across the various views:

$$S_{xy} = \frac{1}{L} \sum_{l=1}^L I(C_x^l = C_y^l) \quad (6)$$

where  $L$  represents the total number of connected components across all views and granularities, and  $I(C_x^l = C_y^l)$  is an indicator function that equals 1 if samples  $x$  and  $y$  belong to the same connected component  $C$  in the  $l$ -th component, and 0 otherwise.

This comprehensive approach, based on the CNN graph, effectively captures multi-scale information from all views. Combined with consensus similarity calculation, it enhances the similarity matrix's ability to distinguish between clusters. The pseudocode is shown in Algorithm 2.

---

#### Algorithm 1: SC-CNN

---

**Input:** Dataset  $X$ , granularity  $k$ , number of clusters  $NC$ .

**Output:** Cluster labels  $CL$ .

- 1 Construct the CNN graph from  $X$  using Eq. (2).
  - 2 Extract connected components  $C_k$  at granularity  $k$ .
  - 3 Train an MLP with one hidden layer on  $X$  using  $C_k$ , where the forward propagation is defined as in Eq. (3).
  - 4 Obtain the low-dimensional embeddings  $Z$  from the hidden layer using Eq. (4).
  - 5 Reconstruct the CNN graph using the learned embeddings  $Z$ .
  - 6 Perform spectral clustering on the reconstructed CNN graph to obtain final cluster labels  $CL$ .
- 

---

#### Algorithm 2: MVSC-CNN

---

**Input:** multi-view dataset  $X = \{X_{v_i}\}$ ,  $NC$ .

**Output:** Cluster labels  $CL$ .

- 1 **for** each view  $v_i$  in  $X$
  - 2 Construct the CNN graph for  $X_{v_i}$  using Eq. (2).
  - 3 Extract connected components at all granularities.
  - 4 **end for**
  - 5 Aggregate all connected components from all views across all granularity levels by Eq. (5).
  - 6 Compute the consensus similarity matrix  $S$  by Eq. (6).
  - 7 Perform spectral clustering on  $S$  to obtain final cluster labels  $CL$ .
-

## Complexity Analysis

The time complexity of Algorithm 1 is determined by several key steps. Constructing the CNN graph (Step 1) and re-constructing it (Step 5) each have a complexity of  $O(n \log n)$ . Extracting connected components (Step 2) operates with  $O(n)$  complexity. Training the MLP (Step 3), with the hidden layer dimensionality fixed at a constant, results in a complexity of  $O(nd)$ . However, the most computationally intensive step is spectral clustering (Step 6), which has a complexity of  $O(n^3)$  due to the eigendecomposition of the similarity matrix. Therefore, the overall time complexity of Algorithm 1 is dominated by the spectral clustering step, leading to a final complexity of approximately  $O(n^3)$ . The time complexity of Algorithm 2, which handles multi-view datasets with  $v_{\max}$  views, is driven by several key steps. Constructing the CNN graph for each view (Step 2) has a complexity of  $O(v_{\max} \cdot n \log n)$ . Extracting connected components (Step 3) and aggregating them (Step 5) across all views and granularities contribute a complexity of  $O(v_{\max} \cdot n)$ . Constructing the consensus similarity matrix (Step 6) adds a complexity of  $O(n^2 \cdot v_{\max})$ . However, the spectral clustering step (Step 7), with a complexity of  $O(n^3)$ , dominates the overall time complexity. Thus, the total time complexity of Algorithm 2 is  $O(n^3)$ .

## Experiment

In this section, we will experimentally validate our SC-CNN and MVSC-CNN. The experiments are divided into two parts: single-view clustering and multi-view clustering.

### Single-View Clustering

#### Algorithms and experimental setup

We evaluated our proposed SC-CNN algorithm against other spectral clustering algorithms that utilize various graph structures, including the k-nearest neighbor (KNN) graph (von Luxburg, 2007), mutual k-nearest neighbor (MKNN) graph (Tan et al., 2020),  $\epsilon$ -neighborhood graph, and Minimum Spanning Tree (MST) graph. Following established norms, the number of neighbors  $K$  for KNN and MKNN graphs was set to  $\ln(n)$  and  $2\ln(n)$ , respectively. These configurations are referred to as (M)KNN-1 and (M)KNN-2, respectively. For the  $\epsilon$  parameter in the  $\epsilon$ -neighborhood graph (Ínkaya, 2015), we adopted the approach from Density Peak Clustering (DPC) (Rodríguez & Laio, 2014) to better capture the manifold structure, setting  $\epsilon$  as a percentile of the distances between all point pairs. This percentile was chosen so that the average number of neighbors within this distance represents approximately 1% and 2% of the total points, referred to as  $\epsilon$ -1 and  $\epsilon$ -2, respectively. For each algorithm based on these graph structures, we ran the experiments 10 times and averaged the results. Additionally, we included comparisons with some of the latest advanced

Datasets	#Samples	#Dimensions	#Clusters
COIL-100	7200	49152	100
COIL-40	2880	49152	40
COIL-20	1440	16384	20
FRGC-v2	2462	3072	20
CMU-PIE	2856	1024	68
JAFFE	213	1024	10

Table 1. Single-view datasets.

graph-based clustering algorithms, such as Graph Structure Fusion-based clustering (GSF) (Zhan, Niu, et al., 2019), Structured bipartite Graph Learning-based scalable subspace clustering (SGL) (Kang et al., 2022), and Consistent and Divergent Graph Clustering (CDGC) (S. Huang et al., 2022). For these three algorithms, we selected the optimal parameter combinations from a wide range of possibilities for each dataset to achieve the highest accuracy. In line with standard practice, we employ two external evaluation metrics: accuracy (ACC) and normalized mutual information (NMI) (Strehl & Ghosh, 2002). ACC measures the proportion of data points correctly classified, while NMI assesses the similarity between the clustering outcomes and the actual data partitions. All experiments were conducted on a workstation with two 14-core Intel Xeon 6132 CPUs clocked at 2.6 GHz and 3.7 GHz with 96 GB memory.

#### Datasets

Our study uses several real-world datasets to benchmark clustering algorithms, selected for their challenging cluster structures and high dimensionality. These datasets include COIL-100 (Nene et al., 1996b), COIL-40, COIL-20 (Nene et al., 1996a), FRGC-v2 (*Face Recognition Grand Challenge (FRGC v.2.0) Data Collection*), CMU-PIE (Sim et al., 2002), and JAFFE (Kanade et al., 2000). Detailed quantitative analysis of these datasets is provided in Table 1.

#### Results

The comparative results of the proposed SC-CNN with other graph structure-based spectral clustering algorithms and graph-based clustering algorithms are presented in Tables 2-3. The proposed SC-CNN outperforms all comparison algorithms on each dataset, across two external validity indices, ACC and NMI. Specifically, for ACC, the proposed SC-CNN surpasses the second-best method by 22.9%, 12.4%, and 10.1% on the CMU-PIE, FRGC-v2, and COIL-20 datasets, respectively. For NMI, on the CMU-PIE and FRGC-v2 datasets, the proposed SC-CNN exceeds the second-best method by 9.7% and 12.1%, respectively.

#### Comparison with deep clustering algorithms

The proposed SC-CNN uses an MLP with a single hidden layer to learn low-dimensional embeddings. Compared to prior deep spectral clustering methods based on deep neural networks, SC-CNN offers better interpretability and training efficiency. In this section, we comprehensively compare SC-CNN with all state-of-the-art (SOTA) deep clustering algorithms that have achieved top performance on the six

Datasets/Methods	KNN-1	KNN-2	MKNN-1	MKNN-2	$\epsilon$ -1	$\epsilon$ -2	MST	CDGC	SGL	GSF	SC-CNN
COIL-100	.6439	.5809	.8101	.7629	.2811	.1957	.5647	<u>.8856</u>	.5553	.7699	<b>.9538</b>
COIL-40	.7610	.6829	.8882	.8507	.1957	.4287	.5537	<u>.8993</u>	.6524	.8424	<b>.9640</b>
COIL-20	.8318	.7394	.8403	.8361	.2236	.1809	.5825	<u>.8771</u>	.7347	.7611	<b>.9778</b>
FRGC-v2	.3710	.3997	.1290	.1632	.1283	.1344	<u>.4119</u>	.3253	.3493	.2750	<b>.5361</b>
CMU-PIE	.7284	.4682	.7368	.5381	.2588	.1728	<u>.7714</u>	.3687	.3326	.4216	<b>1.0000</b>
JAFFE	.9437	<u>.9718</u>	.3103	.7840	.2362	.3521	.9122	.9671	.9249	.8592	<b>.9953</b>

**Table 2.** Comparison of SC-CNN with other spectral clustering algorithms, measured by ACC.

Datasets/Methods	KNN-1	KNN-2	MKNN-1	MKNN-2	$\epsilon$ -1	$\epsilon$ -2	MST	CDGC	SGL	GSF	SC-CNN
COIL-100	.8613	.8212	.9563	.9324	.6764	.5878	.8183	<u>.9774</u>	.8001	.9439	<b>.9897</b>
COIL-40	.9124	.8580	.9668	.9531	.4842	.7765	.7912	<u>.9751</u>	.8447	.9622	<b>.9926</b>
COIL-20	.9000	.8404	<u>.9499</u>	.9259	.3940	.4866	.7654	.9463	.8498	.8948	<b>.9868</b>
FRGC-v2	.5402	<u>.5465</u>	.1230	.1645	.0976	.0986	.5384	.4307	.4548	.3592	<b>.6673</b>
CMU-PIE	.8537	.7295	.8749	.7927	.5645	.4200	<u>.9033</u>	.6259	.5710	.7408	<b>1.0000</b>
JAFFE	.9353	.9577	.4318	.9226	.3032	.4827	.8842	<u>.9639</u>	.9128	.8794	<b>.9918</b>

**Table 3.** Comparison of SC-CNN with other spectral clustering algorithms, measured by NMI.

Datasets	COIL-100	COIL-40	COIL-20	FRGC-v2	CMU-PIE	JAFFE
Rank 1	.947	.922	1	.510	1	.973
	DCV	A-DSSC	JULE	ASC2D	JULE	AdaGAE
Rank 2	.916	.899	.983	.504	1	.967
	JULE	J-DSSC	DSC-FEDL	DDSNnet	DDSNnet	CAN
Rank 3	.863	.842	.975	.470	.979	.944
	DGMM	DSC-FEDL	SADSC	DEPICT	DERC	GAE
Rank 4	.824	.842	.973	.464	.918	.944
	A-DSSC	DSC-DAG	DSC-DAG	DNB	DAutoED	GALA
Rank 5	.796	.840	.968	.461	.883	.908
	J-DSSC	RGRL	S <sup>2</sup> DSCAG	JULE	DEPICT	DFKM
Rank 6	.775	.839	.968	.431	.858	.883
	DBC	S <sup>2</sup> DSCAG	PSOC	MI-ADM	MI-ADM	SpectralNet
Rank 7	.718	.835	.946	.381	.824	.872
	DDSNnet	DASC	DGMM	DPSC	DPSC	MGAE
SC-CNN	<b>.954</b>	<b>.964</b>	.978	<b>.536</b>	1	<b>.995</b>

**Table 4.** Comparison with SOTA deep clustering algorithms from the rankings, measured by ACC.

Datasets	COIL-100	COIL-40	COIL-20	FRGC-v2	CMU-PIE	JAFFE
Rank 1	.985	.967	1	.667	1	.968
	JULE	A-DSSC	JULE	ASC2D	JULE	AdaGAE
Rank 2	.979	.963	.981	.651	1	.964
	DCV	J-DSSC	DSC-FEDL	DNB	DDSNnet	CAN
Rank 3	.946	.951	.979	.610	.996	.933
	A-DSSC	DSC-FEDL	SADSC	DEPICT	DERC	GAE
Rank 4	.943	.928	.974	.580	.970	.925
	J-DSSC	RGRL	S <sup>2</sup> DSCAG	MI-ADM	DAutoED	GALA
Rank 5	.910	.920	.974	.574	.965	.920
	DGMM	DASC	PSOC	JULE	MI-ADM	DFKM
Rank 6	.905	.916	.958	.544	.964	.897
	DBC	DSC-DAG	DSC-DAG	DPSC	DEPICT	MGAE
Rank 7	.886	.913	.910	.522	.925	.870
	DDSNnet	S <sup>2</sup> DSCAG	DGMM	DDSNnet	DPSC	SpectralNet
SC-CNN	<b>.990</b>	<b>.993</b>	.987	<b>.667</b>	1	<b>.992</b>

**Table 5.** Comparison with SOTA deep clustering algorithms from the rankings, measured by NMI.

datasets, as listed in the accuracy rankings on the 'papers\_with\_code'<sup>1</sup> website. These SOTA deep clustering algorithms include: DCV (Wu et al., 2022), JULE (J. Yang et al., 2016), DGMM (J. Wang & Jiang, 2021), A-DSSC (Lim et al., 2020), J-DSSC (Lim et al., 2020), DBC (F. Li et al., 2018), DDSNnet (W. Wang et al., 2021), DSC-FEDL (Q. Huang et al., 2020), DSC-DAG (Yu et al., 2020), RGRL (Kang, Lu, et al., 2020), S<sup>2</sup>DSCAG (Yu et al., 2020), DASC (Zhou et al., 2018), SADSC (Chen et al., 2021), PSOC (J.

Wang et al., 2022), ASC2D (W. Xia et al., 2021), DEPICT (Dizaji et al., 2017), DNB (Z. Wang et al., 2021), MI-ADM (Jabi et al., 2021), DPSC (Hu et al., 2021), DERC (Yan et al., 2020), DautoED (M. Yang & Xu, 2021), AdaGAE (X. Li, Zhang, et al., 2022), CAN (Nie et al., 2014), GAE (Kipf & Welling, 2016), GALA (J. Park et al., 2019), DFKM (R. Zhang et al., 2020), SpectralNet (Shaham et al., 2018a), and MGAE (C. Wang et al., 2017). As shown in Tables 4-5, SC-CNN achieved new SOTA accuracy on five out of the six datasets—COIL-100, COIL-40, FRGC-v2, CMU-PIE, and JAFFE—outperforming all previously reported SOTA deep clustering models.

### Ablation Study

In this section, we explore the impact of two key components within SC-CNN on its performance. The proposed SC-CNN offers two main enhancements over the original SC: CNN graph-based embedding and CNN graph-based similarity. We denote spectral clustering optimized solely with CNN graph-based similarity as SC+CNN-S, and the variant integrating both CNN graph-based similarity and CNN graph-based embedding as SC+CNN-S+CNN-E (i.e., SC-CNN). As shown in Table 6, SC+CNN-S already achieves significant improvements in terms of accuracy (ACC) over the original SC. However, SC+CNN-S+CNN-E (i.e., SC-CNN) outperforms SC+CNN-S on nearly every dataset, underscoring the importance of dual optimization with both components based on the CNN graph.

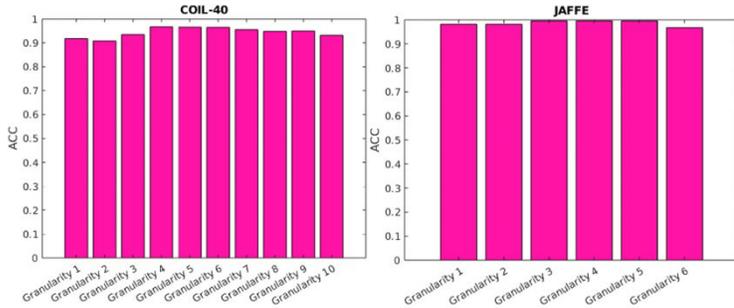
### Parameter Sensitivity

SC-CNN requires only one hyperparameter adjustment: the selection of the granularity level during the training of the CNN-based embedding. This is a significant advantage compared to previous methods, such as AdaGAE (X. Li,

<sup>1</sup> <https://paperswithcode.com/task/image-clustering>

Methods/Datasets	COIL-100	COIL-40	COIL-20	CMU-PIE	FRGC-v2	JAFFE
SC	.6439	.7610	.8318	.7284	.3710	.9437
SC+CNN-S	.9236	.9459	.9229	1	.4617	.9812
SC+CNN-S+CNN-E (SC-CNN)	<b>.9538</b>	<b>.9640</b>	<b>.9778</b>	<b>1</b>	<b>.5361</b>	<b>.9953</b>

**Table 6.** Analysis of the impact of SC-CNN's two components.



**Figure 2.** Sensitivity of SC-CNN to CNN granularity levels.

Zhang, et al., 2022), which requires tuning six hyperparameters, providing greater convenience. In this section, we investigate the sensitivity of SC-CNN’s accuracy (ACC) to different granularity levels on the COIL-40 and JAFFE datasets. As shown in Fig. 2, SC-CNN demonstrates low sensitivity to the choice of granularity levels for the CNN graph.

## Multi-View Clustering

### Algorithms and experimental setup

We compared the proposed MVSC-CNN with 12 advanced multi-view clustering algorithms, 10 of which are based on spectral clustering. These include: MLRSSC (Brbić & Kopriva, 2018), GMC (H. Wang et al., 2020), UGLMC (Liang et al., 2019), CBF-MS (Zheng et al., 2020), V3H (Fang et al., 2020), LMVSC (Kang, Zhou, et al., 2020), AASC (H.-C. Huang et al., 2012), AWP (Nie et al., 2018), CoReg (Kumar et al., 2011), MCGC (Zhan, Nie, et al., 2019), RMSC (R. Xia et al., 2014), and WMSC (Zong et al., 2018). Unlike the proposed MVSC-CNN, all of these methods require hyperparameter tuning. Therefore, the parameters for these algorithms were tuned as recommended in the original papers to achieve the best results. For further details, please refer to the supplementary materials.

### Datasets

We used five widely adopted multi-view datasets to evaluate the performance of the proposed MVSC-CNN against other comparative algorithms. These include two UCI datasets<sup>2</sup>, 100-leaves and UCI-digits, as well as three additional datasets: COIL-20 (Nene et al., 1996a), Handwritten (X. Cai

Datasets	#Views	#Samples	#Dimensions	#Clusters
100-leaves	3	1600	64, 64, 64	100
UCI-digits	3	2000	76, 216, 64	10
COIL-20	3	1440	1024, 3304, 6750	20
Handwritten	2	2000	240, 76	10
UMIST	3	575	30, 30, 30	20

**Table 7.** Multi-view datasets.

Methods	100-leaves	UCI-digits	COIL-20	Handwritten	UMIST
MLRSSC	.7199	.8773	.6497	.7895	.4517
GMC	.8237	.8495	.7910	.8300	.5217
UGLMC	.7881	.8825	.9014	.7425	.6087
CBF-MS	.7823	.9324	.7449	.8945	.4701
V3H	.8237	.9051	.6012	.8669	.5294
LMVSC	.6575	.8935	.7569	.9005	.4696
AASC	.8581	.8505	.7924	.8335	.4348
AWP	.7856	.8670	.6757	.9325	.5217
CoReg	.8456	.9560	.8472	.9110	.5043
MCGC	.6244	.4920	.4389	.1005	.4643
RMSC	.7781	.2495	.3896	.4120	.4539
WMSC	.8769	.8410	.8465	.8335	.4539
<b>MVSC-CNN</b>	<b>.9331</b>	<b>.9715</b>	<b>.9750</b>	<b>.9800</b>	<b>.7183</b>

**Table 8.** Comparison of MVSC-CNN with other multi-view spectral clustering algorithms, measured by ACC.

et al., 2013), and UMIST (J. Yang & Lin, 2023), all of which have been used in previous studies. A complete description of these datasets is provided in Table 7.

## Results

The comparative results of the proposed MVSC-CNN with other advanced multi-view clustering algorithms in terms of ACC are presented in Tables 8. The proposed MVSC-CNN outperforms all comparison algorithms on each dataset. Specifically, MVSC-CNN achieves improvements of 5.6%, 7.4%, and 11.0% over the second-best performing algorithms on the 100-leaves, COIL-20, and UMIST datasets, respectively. Notably, MVSC-CNN, which does not require any trade-off parameter tuning, consistently outperforms prior parameter-dependent algorithms across all five datasets.

## Conclusion

In this paper, we addressed key limitations in traditional spectral clustering by introducing a novel hierarchical similarity graph, the Constrained 1-Nearest Neighbor (CNN) graph. This graph effectively captures both local and global nonlinear structures across varying levels of granularity. Based on this, we developed two enhanced spectral clustering algorithms: SC-CNN for single-view data and MVSC-CNN for multi-view data. SC-CNN improves generalization by leveraging cluster-friendly low-dimensional embeddings and a more robust similarity graph, while MVSC-CNN integrates multi-granularity information from multiple views to further enhance clustering performance. Our extensive experiments on real-world datasets demonstrate that both SC-CNN and MVSC-CNN consistently outperform state-of-the-art methods, offering superior accuracy and robustness. These findings highlight the effectiveness of our proposed CNN graph in addressing the complexities of real-world data and provide a promising direction for further improvements in spectral clustering.

<sup>2</sup> <https://archive.ics.uci.edu/ml/datasets>

## References

- Affeldt, S., Labiod, L., & Nadif, M. (2020). Spectral clustering via ensemble deep autoencoder learning (SC-EDAE). *Pattern Recognition*, 108, 107522.
- Alshammari, M., Stavarakakis, J., & Takatsuka, M. (2021). Refining a k-nearest neighbor graph for a computationally efficient spectral clustering. *Pattern Recognition*, 114, 107869.
- Brbić, M., & Kopriva, I. (2018). Multi-view low-rank sparse subspace clustering. *Pattern Recognition*, 73, 247–258.
- Cai, X., Nie, F., & Huang, H. (2013). Multi-View K-Means Clustering on Big Data. *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, 2598–2604.
- Cai, Y., Huang, J. Z., & Yin, J. (2022). A new method to build the adaptive k-nearest neighbors similarity graph matrix for spectral clustering. *Neurocomputing*, 493, 191–203.
- Chen, Z., Ding, R., Chin, T.-W., & Marculescu, D. (2019). Understanding the Impact of Label Granularity on CNN-based Image Classification (arXiv:1901.07012).
- Chen, Z., Ding, S., & Hou, H. (2021). A novel self-attention deep subspace clustering. *International Journal of Machine Learning and Cybernetics*, 12(8), 2377–2387.
- Dizaji, K. G., Herandi, A., Deng, C., Cai, W., & Huang, H. (2017). Deep Clustering via Joint Convolutional Autoencoder Embedding and Relative Entropy Minimization, *IEEE International Conference on Computer Vision (ICCV)* 5747–5756.
- Face Recognition Grand Challenge (FRGC v.2.0) data collection. <https://cvrl.nd.edu/projects/data/#face-recognition-grand-challenge-frgc-v20-data-collection>
- Fang, X., Hu, Y., Zhou, P., & Wu, D. O. (2020). V3H: View Variation and View Heredity for Incomplete Multiview Clustering. *IEEE Transactions on Artificial Intelligence*, 1(03), 233–247.
- Hu, W., Chen, C., Ye, F., Zheng, Z., & Du, Y. (2021). Learning deep discriminative representations with pseudo supervision for image clustering. *Information Sciences*, 568, 199–215.
- Huang, D., Wang, C.-D., & Lai, J.-H. (2023). Fast Multi-View Clustering Via Ensembles: Towards Scalability, Superiority, and Simplicity. *IEEE Transactions on Knowledge and Data Engineering*, 35(11), 11388–11402.
- Huang, D., Wang, C.-D., Peng, H., Lai, J., & Kwok, C.-K. (2021). Enhanced Ensemble Clustering via Fast Propagation of Cluster-Wise Similarities. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 51(1), 508–520.
- Huang, D., Wang, C.-D., Wu, J.-S., Lai, J.-H., & Kwok, C.-K. (2020). Ultra-Scalable Spectral Clustering and Ensemble Clustering. *IEEE Transactions on Knowledge and Data Engineering*, 32(6), 1212–1226.
- Huang, H.-C., Chuang, Y.-Y., & Chen, C.-S. (2012). Affinity aggregation for spectral clustering, *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 773–780.
- Huang, Q., Zhang, Y., Peng, H., Dan, T., Weng, W., & Cai, H. (2020). Deep subspace clustering to achieve jointly latent feature extraction and discriminative learning. *Neurocomputing*, 404, 340–350.
- Huang, S., Ota, K., Dong, M., & Li, F. (2019). MultiSpectralNet: Spectral Clustering Using Deep Neural Network for Multi-View Data. *IEEE Transactions on Computational Social Systems*, 6(4), 749–760.
- Huang, S., Tsang, I. W., Xu, Z., & Lv, J. (2022). Measuring Diversity in Graph Learning: A Unified Framework for Structured Multi-View Clustering. *IEEE Transactions on Knowledge and Data Engineering*, 34(12), Article 12.
- İnkaya, T. (2015). A parameter-free similarity graph for spectral clustering. *Expert Systems with Applications*, 42(24), 9489–9498.
- İnkaya, T., Kayaligil, S., & Özdemirel, N. E. (2015). An adaptive neighbourhood construction algorithm based on density and connectivity. *Pattern Recognition Letters*, 52, 17–24.
- Jabi, M., Pedersoli, M., Mitiche, A., & Ayed, I. B. (2021). Deep Clustering: On the Link Between Discriminative Models and K-Means. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(06), 1887–1896.
- Jianbo Shi, & Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8), Article 8.
- Kanade, T., Cohn, J. F., & Tian, Y. (2000). Comprehensive database for facial expression analysis. *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, 46–53.
- Kang, Z., Lin, Z., Zhu, X., & Xu, W. (2022). Structured Graph Learning for Scalable Subspace Clustering: From Single View to Multiview. *IEEE Transactions on Cybernetics*, 52(9), Article 9.
- Kang, Z., Lu, X., Liang, J., Bai, K., & Xu, Z. (2020). Relation-Guided Representation Learning. *Neural Networks*, 131, 93–102.
- Kang, Z., Shi, G., Huang, S., Chen, W., Pu, X., Zhou, J. T., & Xu, Z. (2020). Multi-graph fusion for multi-view spectral clustering. *Knowledge-Based Systems*, 189, 105102.
- Kang, Z., Zhou, W., Zhao, Z., Shao, J., Han, M., & Xu, Z. (2020). Large-scale Multi-view Subspace Clustering in Linear Time, *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04), 4412–4419.
- Kipf, T. N., & Welling, M. (2016). Variational Graph Auto-Encoders (arXiv:1611.07308).
- Kumar, A., Rai, P., & Daume, H. (2011). Co-regularized Multi-view Spectral Clustering. *Advances in Neural Information Processing Systems*, 24.
- Law, M. T., Urtasun, R., & Zemel, R. S. (2017). Deep Spectral Clustering Learning. *Proceedings of the 34th International Conference on Machine Learning*, 1985–1994.
- Li, F., Qiao, H., & Zhang, B. (2018). Discriminatively boosted image clustering with fully convolutional auto-encoders. *Pattern Recognition*, 83, 161–173.

- Li, X., Wei, T., & Zhao, Y. (2022). Deep Spectral Clustering With Constrained Laplacian Rank. *IEEE Transactions on Neural Networks and Learning Systems*, 1–12.
- Li, X., Zhang, H., & Zhang, R. (2022). Adaptive Graph Auto-Encoder for General Data Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12), 9725–9732.
- Liang, Y., Huang, D., & Wang, C.-D. (2019). Consistency Meets Inconsistency: A Unified Graph Learning Framework for Multi-view Clustering. 2019 IEEE International Conference on Data Mining (ICDM), 1204–1209.
- Lim, D., Vidal, R., & Haeffele, B. (2020). Doubly Stochastic Subspace Clustering. arXiv:2011.14859
- Lu, H., Chen, C., Wei, H., Ma, Z., Jiang, K., & Wang, Y. (2022). Improved deep convolutional embedded clustering with re-selectable sample training. *Pattern Recognition*, 127, 108611.
- Mahon, L., & Lukasiewicz, T. (2021). Selective Pseudo-Label Clustering. In S. Edelkamp, R. Möller, & E. Rueckert (Eds.), *KI 2021: Advances in Artificial Intelligence* (pp. 158–178).
- Nene, S. A., Nayar, S. K., & Murase, H. (1996a). Columbia Object Image Library (COIL-20). Technical Report, Technical Report CUCS-005-96.
- Nene, S. A., Nayar, S. K., & Murase, H. (1996b). Columbia Object Image Library (COIL-100). Technical Report, CUCS-006-96.
- Ng, A. Y., Jordan, M. I., & Weiss, Y. (2002). On Spectral Clustering: Analysis and an algorithm. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in Neural Information Processing Systems 14* (pp. 849–856). MIT Press.
- Nie, F., Tian, L., & Li, X. (2018). Multiview Clustering via Adaptively Weighted Procrustes. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2022–2030.
- Nie, F., Wang, X., & Huang, H. (2014). Clustering and projected clustering with adaptive neighbors. *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 977–986.
- Park, J., Lee, M., Chang, H. J., Lee, K., & Choi, J. Y. (2019). Symmetric Graph Convolutional Autoencoder for Unsupervised Graph Representation Learning. 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 6518–6527.
- Park, S., Han, S., Kim, S., Kim, D., Park, S., Hong, S., & Cha, M. (2021). Improving Unsupervised Image Clustering With Robust Learning, 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 12278–12287.
- Ren, Y., Pu, J., Cui, C., Zheng, Y., Chen, X., Pu, X., & He, L. (2024). Dynamic Weighted Graph Fusion for Deep Multi-View Clustering. *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, 4842–4850.
- Rodriguez, A., & Laio, A. (2014). Clustering by fast search and find of density peaks. *Science*, 344(6191), Article 6191.
- Shaham, U., Stanton, K., Li, H., Nadler, B., Basri, R., & Kluger, Y. (2018a). SpectralNet: Spectral Clustering using Deep Neural Networks (arXiv:1801.01587).
- Sim, T., Baker, S., & Bsat, M. (2002). The CMU Pose, Illumination, and Expression (PIE) database. *Proceedings of Fifth IEEE International Conference on Automatic Face Gesture Recognition*, 53–58.
- Strehl, A., & Ghosh, J. (2002). Cluster Ensembles—A Knowledge Reuse Framework for Combining Multiple Partitions. *Journal of Machine Learning Research*, 3(Dec), 583–617.
- Tan, M., Zhang, S., & Wu, L. (2020). Mutual kNN based spectral clustering. *Neural Computing and Applications*, 32(11), 6435–6442.
- Tang, C., Li, Z., Wang, J., Liu, X., Zhang, W., & Zhu, E. (2023). Unified One-Step Multi-View Spectral Clustering. *IEEE Transactions on Knowledge and Data Engineering*, 35(6), 6449–6460.
- von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and Computing*, 17(4), 395–416.
- Wang, C., Pan, S., Long, G., Zhu, X., & Jiang, J. (2017). MGAE: Marginalized Graph Autoencoder for Graph Clustering. *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 889–898.
- Wang, H., Yang, Y., & Liu, B. (2020). GMC: Graph-Based Multi-View Clustering. *IEEE Transactions on Knowledge and Data Engineering*, 32(6), 1116–1129.
- Wang, J., & Jiang, J. (2021). Unsupervised deep clustering via adaptive GMM modeling and optimization. *Neurocomputing*, 433, 199–211.
- Wang, J., Wang, L., & Jiang, J. (2022). Preserving similarity order for unsupervised clustering. *Pattern Recognition*, 128, 108670.
- Wang, S., Zhang, L., Chen, W., Wang, F., & Li, H. (2022). Refining pseudo labels for unsupervised Domain Adaptive Re-Identification. *Knowledge-Based Systems*, 242, 108336.
- Wang, W., Chen, F., Ge, Y., Huang, S., Zhang, X., & Yang, D. (2021). Discriminative deep semi-nonnegative matrix factorization network with similarity maximization for unsupervised feature learning. *Pattern Recognition Letters*, 149, 157–163.
- Wang, Z., Ni, Y., Jing, B., Wang, D., Zhang, H., & Xing, E. (2021). DNB: A Joint Learning Framework for Deep Bayesian Nonparametric Clustering. *IEEE Transactions on Neural Networks and Learning Systems*, 1–11.
- Wen, J., Sun, H., Fei, L., Li, J., Zhang, Z., & Zhang, B. (2021). Consensus guided incomplete multi-view spectral clustering. *Neural Networks*, 133, 207–219.
- Wen, J., Xu, Y., & Liu, H. (2020). Incomplete Multiview Spectral Clustering With Adaptive Graph Learning. *IEEE Transactions on Cybernetics*, 50(4), 1418–1429.
- Wu, L., Yuan, L., Zhao, G., Lin, H., & Li, S. Z. (2022). Deep Clustering and Visualization for End-to-End High-Dimensional Data Analysis. *IEEE Transactions on Neural Networks and Learning Systems*, 1–12.

- Xia, R., Pan, Y., Du, L., & Yin, J. (2014). Robust Multi-View Spectral Clustering via Low-Rank and Sparse Decomposition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 28(1), Article 1.
- Xia, W., Zhang, X., Gao, Q., & Gao, X. (2021). Adversarial self-supervised clustering with cluster-specificity distribution. *Neurocomputing*, 449, 38–47.
- Yan, Y., Hao, H., Xu, B., Zhao, J., & Shen, F. (2020). Image Clustering via Deep Embedded Dimensionality Reduction and Probability-Based Triplet Loss. *IEEE Transactions on Image Processing*, 29, 5652–5661.
- Yang, J., & Lin, C.-T. (2023). Multi-View Adjacency-Constrained Hierarchical Clustering. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 7(4), 1126–1138.
- Yang, J., & Lin, C.-T. (2024). Enhanced Adjacency-Constrained Hierarchical Clustering Using Fine-Grained Pseudo Labels. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 8(3), 2481–2492.
- Yang, J., Parikh, D., & Batra, D. (2016). Joint Unsupervised Learning of Deep Representations and Image Clusters. 2016 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5147–5156.
- Yang, M., & Xu, S. (2021). Orthogonal Nonnegative Matrix Factorization using a novel deep Autoencoder Network. *Knowledge-Based Systems*, 227, 107236.
- Yang, X., Deng, C., Zheng, F., Yan, J., & Liu, W. (2019). Deep Spectral Clustering Using Dual Autoencoder Network. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4066–4075.
- Yu, Z., Zhang, Z., Cao, W., Liu, C., Philip Chen, J., & Wong, H. S. (2020). GAN-based Enhanced Deep Subspace Clustering Networks. *IEEE Transactions on Knowledge and Data Engineering*, 1–1.
- Zelnik-Manor, L., & Pietro, P. (2004). Self-tuning spectral clustering. *Advances in neural information processing systems* 17.
- Zhan, K., Nie, F., Wang, J., & Yang, Y. (2019). Multiview Consensus Graph Clustering. *IEEE Transactions on Image Processing*, 28(3), 1261–1270.
- Zhan, K., Niu, C., Chen, C., Nie, F., Zhang, C., & Yang, Y. (2019). Graph Structure Fusion for Multiview Clustering. *IEEE Transactions on Knowledge and Data Engineering*, 31(10), Article 10.
- Zhang, H., Nie, F., & Li, X. (2023). Large-Scale Clustering With Structured Optimal Bipartite Graph. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(8), 9950–9963.
- Zhang, R., Li, X., Zhang, H., & Nie, F. (2020). Deep Fuzzy K-Means With Adaptive Loss and Entropy Regularization. *IEEE Transactions on Fuzzy Systems*, 28(11), 2814–2824.
- Zhang, X., Ge, Y., Qiao, Y., & Li, H. (2021). Refining Pseudo Labels With Clustering Consensus Over Generations for Unsupervised Object Re-Identification. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3436–3445.
- Zhang, X., Li, J., & Yu, H. (2011). Local density adaptive similarity measurement for spectral clustering. *Pattern Recognition Letters*, 32(2), 352–358.
- Zhao, Y., & Li, X. (2023a). Deep Spectral Clustering With Regularized Linear Embedding for Hyperspectral Image Clustering. *IEEE Transactions on Geoscience and Remote Sensing*, 61, 1–11.
- Zhao, Y., & Li, X. (2023b). Spectral Clustering With Adaptive Neighbors for Deep Learning. *IEEE Transactions on Neural Networks and Learning Systems*, 34(4), 2068–2078.
- Zheng, Q., Zhu, J., Tian, Z., Li, Z., Pang, S., & Jia, X. (2020). Constrained bilinear factorization multi-view subspace clustering. *Knowledge-Based Systems*, 194, 105514.
- Zhou, P., Hou, Y., & Feng, J. (2018). Deep Adversarial Subspace Clustering. 2018 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1596–1604.
- Zong, L., Zhang, X., Liu, X., & Yu, H. (2018). Weighted Multi-View Spectral Clustering Based on Spectral Perturbation. *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32, No. 1.