EAI Endorsed Transactions
on Scalable Information Systems

Research Article EAI.EU

A novel image clustering method based on coupled convolutional and graph convolutional network

Rangjun Li[1,*]

[1]School of Electronic and Electrical Engineering, Zhengzhou University of Science and Technology, Zhengzhou 450064 China

Abstract

Image clustering is a key and challenging task in the field of machine learning and computer vision. Technically, image clustering is the process of grouping images without the use of any supervisory information in order to retain similar images within the same cluster. This paper proposes a novel image clustering method based on coupled convolutional and graph convolutional network. It solves the problem that the deep clustering method usually only focuses on the useful features extracted from the sample itself, and seldom considers the structural information behind the sample. Experimental results show that the proposed algorithm can effectively extract more discriminative deep features, and the model achieves good clustering effect due to the combination of attribute information and structure information of samples in GCN.

Keywords: machine learning, image clustering, coupled convolutional, graph convolutional network.

Received on 04 November 2021, accepted on 14 November 2021, published on 16 November 2021

doi: 10.4108/eai.16-11-2021.172132

*Corresponding author. Email: 352720214@qq.com

1. Introduction

Image clustering is very important in computer vision. In order to make full use of these unlabeled data and study the correlation between them, many clustering algorithms have been proposed and successfully applied in various practical applications, such as image segmentation [1-2], target detection [3] and image classification [4,5]. Among them, traditional clustering methods, such as K-means clustering algorithm [6], spectral clustering (SC) algorithm [7] and non-negative matrix factorization clustering (NMF) algorithm [8], capture similarity based on the concept of distance in the original data space, so they are considered as shallow models. Although the shallow models have been successfully applied in a variety of scenarios, calculating distance-based measures in raw data space is only suitable for describing local relationships in the data space and is limited in expressing potential dependencies between inputs, which is insufficient to discover semantic similarity.

With the booming development of deep learning, many researchers have shifted their attention to deep unsupervised feature learning and clustering [9-12]. Thus, a new clustering strategy, called deep clustering, emerged. When dealing with large, high-semantic and high-dimensional data, the multi-layer architecture based on deep neural network unsupervised representation learning has become the natural choice. In addition, deep clustering combines prior knowledge with clustering to obtain the optimal embedded subspace for clustering. Compared with traditional clustering methods, deep clustering method can effectively simulate the input distribution and capture the nonlinear characteristics of the input. Therefore, it can well solve the limitations of

shallow model and is more suitable for practical clustering scenarios.

The deep clustering method integrates the clustering target into the powerful representation capability of deep learning. Therefore, learning the effective feature representation directly determines the quality of clustering. In order to make potential representations more discriminative, most existing deep clustering methods attempt to minimize reconstruction losses. For example, Xie et al. [13] used clustering loss to help auto-encoder learn data representation with high clustering cohesion. Bashon et al. [14] used variational autoencoder to learn better data representation of clustering.

Although deep learning has achieved great success in many important tasks, there are still several problems when using deep neural networks to perform clustering tasks. First, many authors try to combine mature clustering algorithms with deep learning. For example, network training is combined with k-means goals [15-17]. However, a simple combination of clustering and presentation learning methods often leads to regression and resolution. Secondly, auto-encoders are widely used in deep clustering and only consider the reconstructed feature representation, lacking discriminant ability. The ideal approach would be to train a discriminator with adversarial networks, but this further increases the difficulty of the task [18]. In order to learn more discriminative deep features, Chen et al. [19] mined the similarity information contained in image triples. Hjelm et al. [20] maximized the mutual information between features. Finally, most deep clustering only focuses on the characteristics of the data itself and seldom considers the structural information between the data, which can often reveal the potential similarity between samples, thus providing valuable guidance for learning representation. Abdella et al. [21] connected stack auto-encoder with graph convolutional neural network (GCN) through transfer operator, and used self-supervision mechanism to optimize feature extraction and clustering training process. Although structural information plays an important role in data representation learning, it is seldom used in deep clustering.

To solve these problems, a novel image clustering method based on coupled convolutional and graph convolutional network (CCGCN) is proposed in this paper. The embedded mutual information estimation network and minimized prior distribution constraint in the convolutional auto-encoder. Then, the sample's own attribute information learned from the deep auto-coding network is integrated into the graph convolutional neural network, realizing the collaborative learning of the sample's own attribute information and structure information, and completing the end-to-end clustering task, which effectively improves the feature discrimination ability while retaining more available information. Finally, experiments are carried out on three classical image data sets to verify the effectiveness of the proposed algorithm.

2. Related works

2.1. Deep clustering based on auto-encoder

The clustering method based on a auto-encoder (AE) relies on a joint execution to represent a linear combination of learning and clustering of two objective functions. The joint optimization process is described as:

$$L = L_{res} + \gamma L_c \tag{1}$$

Where L_{res} is a function of reconstructed loss. L_c is embedded cluster loss. γ is a super-parameter, which is a factor that controls the degree of distortion in the embedded space. The general network architecture of the AE-based deep clustering algorithm is shown in figure 1, where X is the input image and the $\hat{X}$ is the reconstructed image.

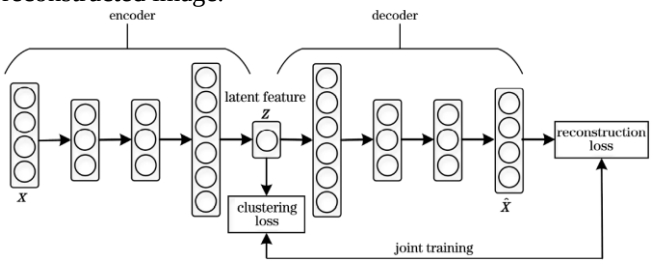

Figure 1. Architecture of AE

2.2. Deep clustering based on variational auto-encoder

The AE-based deep clustering approach has been improved significantly compared to traditional clustering methods. However, they are specifically designed for clustering and do not reveal the true underlying structure of the sample. In addition, assumptions based on dimensional reduction techniques are usually independent of those of clustering technologies, so there is no theory to ensure that the network can learn viable representations. The variational auto-encoder (VAE) as a kind of deep degree generated module type, can be considered as a AE generated variant, it exerts a priori probability distribution characteristics of potential said, which will become bayesian approach combined with flexibility and scalability of the neural network, using the variational lower weight parameterized by a differentiable lower unbiased estimator. The objective function of the depth

clustering algorithm based on the variational auto-encoder is expressed as:

$$L(\theta,\varphi;X)=\sum_{i}^{N}\{-D_{KL}[p_\theta(z\mid x_i)\parallel p(z)]+E_{p_\theta(z|x_i)}[\log_2 q_\varphi(x_i\mid z)]\} \quad (2)$$

Where p(z) is the prior distribution of the whole potential feature space. $p_\theta(z\mid x_i)$ is a conditional posterior distribution. $q_\varphi(x_i\mid z)$ is the likelihood function. N is the total number of samples. $D_{KL}(\cdot)$ is the Kullback-Leibler (k-L) divergence between the conditional posterior distribution $p_\theta(z\mid x_i)$ and the prior distribution p(z) of the entire potential feature space. E() is the expectation of the function. The general network architecture based on VAE deep clustering algorithm is shown in figure 2, where $N(\mu_c,\sigma_c)$ is a normal distribution.

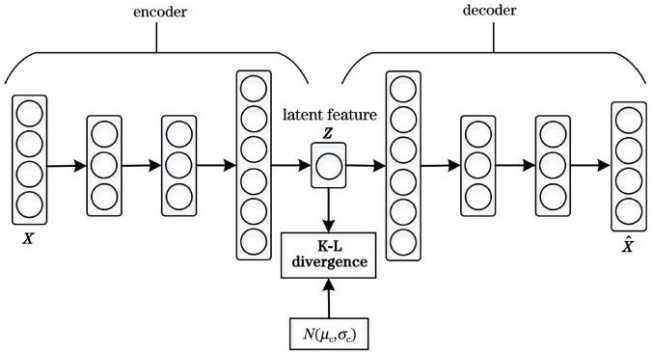

Figure 2. General network architecture of deep clustering algorithm based on VAE

2.3. Deep clustering based on graph convolutional neural network

These deep clustering methods generally focus only on data representations learned from the sample itself, while another important message of learning characterization, namely, the structural information of the data, is rarely taken into account. In order to manage the structural information behind the data, the clustering method based on GCN [22] has been widely used. Kipf et al. [23] presented the graph auto-encoder (GAE) and graph sub-encoder (VGAE), which used graph convolution as an encoder to integrate the graph structure into the node characteristics and learned the embedding of the node. However, the vast majority of GCN-based clustering methods rely on reconstructing the adjacent matrix, which can only learn data representation from the graph structure, while ignoring the data itself characteristics.

3. Proposed CCGCN model

We propose a CCGCN model for extracting and processing image-related features from different image blocks as shown in figure 3. This model mainly consists of two parts: image block level feature extraction network and b-GCN

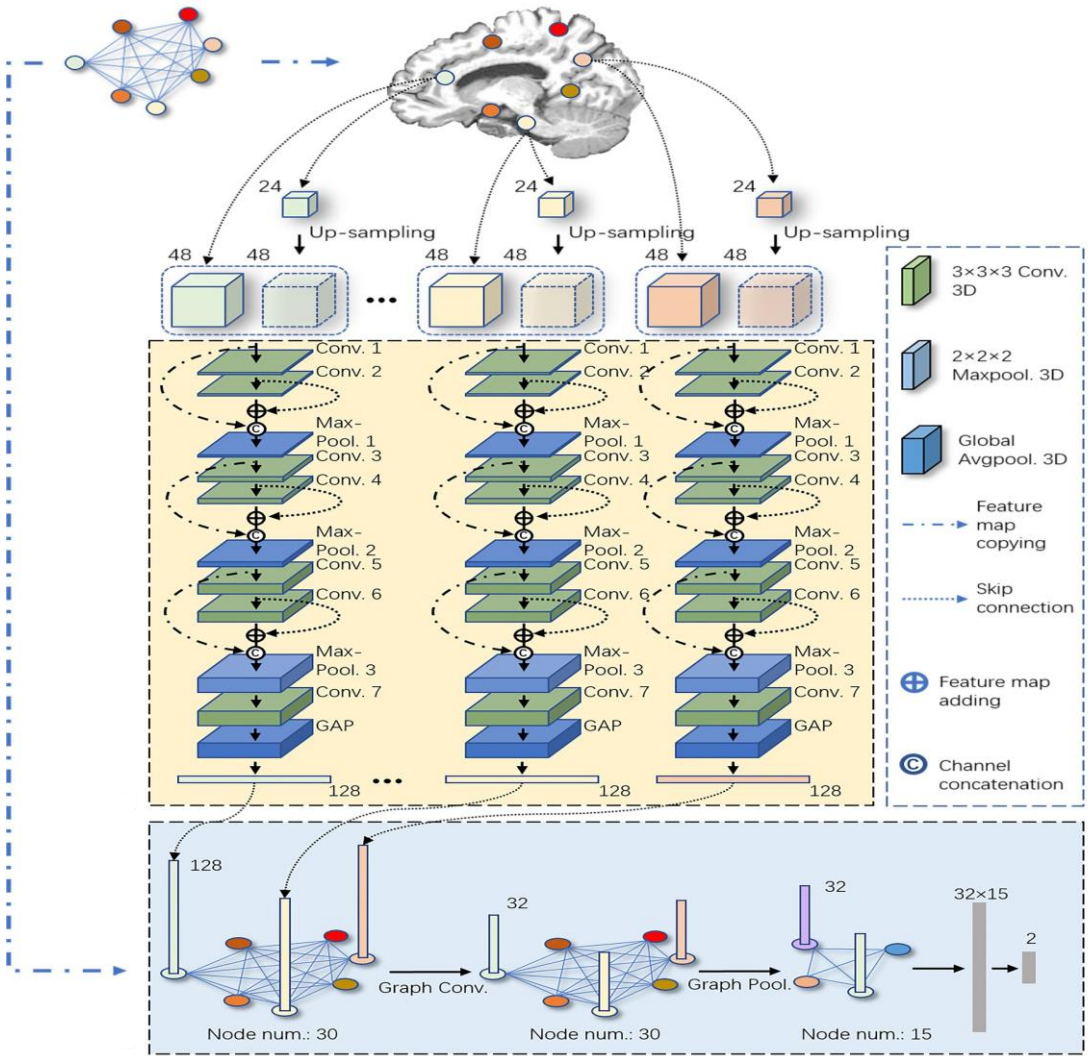

Figure 3. The proposed CCGCN model.

1) Image block hierarchical feature extraction network. Its function is to extract features from corresponding image blocks. The input of the network is two image blocks of the same size and different scales, and the output is tag related feature vector. In order to generate tag-related features, the network uses seven convolutional layers to extract features from the input image blocks, and batch standardized operation and linear rectification unit (ReLU) are added after each convolutional layer [24]. In order to promote feature communication between layers, avoid gradient disappearance in the process of optimization, and enhance feature extraction capability of the network, residual connection and dense connection are adopted in the inter-layer connection of the network. The last feature image output from the convolutional layer is compressed into a vector by the global average pooling layer, and then features are further integrated by the full connection layer. Finally, feature vectors related to disease labels are output (the dimension of feature vectors extracted from each image block in this experiment is defined as 128 dimensions).

2) b-GCN: Graph structure G(V,E) can be constructed by using the spatial structure of feature points mentioned above.

Where V is the node set of graph structure. E is the set of edges between nodes in the graph structure. The feature points are defined as the nodes of the graph. The Euclidean distance between nodes is defined as the edge of the graph. μ -dimensional feature vectors (μ =128) extracted from image blocks at each feature point (denoted as K, and there are 30 feature points in total, K=30) are defined as feature vectors at corresponding nodes.

Thus, we can construct the node characteristic matrix $H \in R^{K\times\mu}$.

$$H = \begin{bmatrix} (\lambda^1)^T \\ (\lambda^2)^T \\ \vdots \\ (\lambda^K)^T \end{bmatrix} \tag{3}$$

Meanwhile, in order to describe the edge in the graph structure, namely, the adjacency state of the node, $d_{i,j}$ is defined as the Euclidean distance between the i-th and j-th feature points. Then the adjacent short matrix $A \in R^{K \times K}$ can be expressed as:

$$A = \begin{bmatrix} 1 & \frac{1}{d_{1,2}} & \cdots & \frac{1}{d_{1,K}} \\ \frac{1}{d_{2,1}} & 1 & \cdots & \frac{1}{d_{2,K}} \\ \vdots & & \ddots & \vdots \\ \frac{1}{d_{1,2}} & \frac{1}{d_{1,2}} & \cdots & 1 \end{bmatrix} \tag{4}$$

By using the feature matrix and adjacency matrix of nodes, we can describe the topological structure of the image by the distribution of feature points in the image, and process it by b-GCN. For graph convolution layer of layer l, the feature matrix H^{l+1} output by graph convolution operation can be expressed as:

$$H^{l+1} = \mathrm{Re}\,LU(A^l H^l W^l) \tag{5}$$

Where $W^l \in R^{\mu \times \mu'}$ represents the coefficient matrix of graph convolution, including $\mu \times \mu'$ learnable parameters (μ' =32 in this experiment). ReLU() is the linear rectification unit and provides nonlinear operation for graph convolution operation. In addition, in order to further carry out adaptive modeling of topology space, we add a graph pooling layer for b-GCN by referring to the work of Ying et al. [25]. The graph pooling operation is realized by pooling matrix $S \in R^{K \times K'}$, where K=15. For the l-th pooling layer, the adjacency matrix A_{pool}^{l+1} and the feature matrix H_{pool}^{l+1} after pooling can be expressed as:

$$A_{pool}^{l+1} = S^T A^l S \tag{6}$$

$$H_{pool}^{l+1} = S^T H^l \tag{7}$$

The feature matrix output by b-GCN is stretched into a 1-dimensional vector, and the prediction results of image labels are obtained after a fully connection layer processing.

The auto-coding network can learn useful representations from the sample, such as $Z^{(1)}, Z^{(2)}, \cdots, Z^{(L)}$. But the structural information between samples is ignored. In order to obtain the structural information of the input sample, a K-nearest Neighbor (KNN) graph is constructed for the original sample, and then the transfer operator δ is used to integrate the feature representation of each layer learned by the GCN module into each layer corresponding to the GCN module. At this point, GCN module can simultaneously learn the attribute information of the sample itself and the structural information between the samples. For the weight matrix W, the representation learning $H^{(l)}$ of the *l*-th layer of GCN module can be obtained through convolution operation. The standard graph convolution layer propagation formula is defined as:

$$H^{(l)} = \sigma(\tilde{D}^{-0.5}\tilde{A}\tilde{D}^{-0.5}H^{(l-1)}W^{(l-1)}) \tag{8}$$

Where $\tilde{A} = A + I$, $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$. I is the unit diagonal matrix of the adjacency matrix A of each node. $\tilde{D}^{-0.5}\tilde{A}\tilde{D}^{-0.5}$ is the normalized adjacency matrix.

In order to simultaneously learn the structural information and samples in the GCN module. For the attribute information of the sample itself, the transfer operator δ is used to weighted sum the two representations, i.e.,

$$\tilde{H}^{(l-1)} = (1-\delta)H^{(l-1)} + \delta Z^{(l-1)} \tag{9}$$

In this way, b-GCN module and GCN module are connected layer by layer. By combining equations (8) and (9), we can get:

$$H^{(l)} = \sigma(\tilde{D}^{-0.5}\tilde{A}\tilde{D}^{-0.5}H^{(l-1)}W^{(l-1)}) \tag{10}$$

In this way, the representations learned by the first layer of the deep auto-coding network will be integrated into the corresponding layer of the GCN module for dissemination. In the last layer of GCN module, the multi-classification layer with Softmax function is used to output Y, so as to predict the distribution of samples.

$$Y = Soft\max(\tilde{D}^{-0.5}\tilde{A}\tilde{D}^{-0.5}H^{(l)}W^{(L)}) \tag{11}$$

Where Y is a probability distribution.

4. Experimental results and analysis

The experiment is divided into three parts. The proposed algorithms and six other clustering algorithms are first compared on three classic data sets including USPS, MNIST, and Fashion-MNIST. The clustering performance of the CCGCN algorithm is evaluated on two quantitative measures, Cluster Accuracy (ACC) and Normalization Mutual Information (NMI). Then, by controlling several influencing factors, a series of ablation experiments are carried out to verify the validity of the proposed algorithm. Finally, the effect of different values

of CCGCN algorithm parameters λ_1 and λ_2 on clustering performance is also discussed.

4.1. Dataset

To demonstrate that the CCGCN algorithm is better able to handle a variety of types of data sets, three classic image datasets (USPS, MNIST, and Fashion-MNIST) are selected for experimentation [26]. Because the clustering task is completely unsupervised, the training samples are stitched together with the test samples in the experiment. Statistics for these datasets are shown in table 1.

Table 1. Dataset explanation

Dataset	Sample number	Class number	Dimension
USPS	9298	10	1×16×16
MNIST	70000	10	1×28×28
Fashion-MNIST	70000	10	1×28×28

The USPS (U.S. Postal Service's handwritten digital) dataset consists of grayscale digital handwritten images (16×16 pixels), including 9298 images, 10 categories. 4649 are as training samples and 4649 images are as testing samples. The MNIST handwriting dataset consists of 28×28 pixel grayscale digital handwritten images, consisting of 70000 images in 10 categories. 60000 images are as training set, 10000 images are as testing set. The Fashion-MNIST dataset covers a total of 70000 fashion products of all types from 10 categories with 60000 images as training set and 10000 images as test set. Figure 4 (a) and (b) give some sample examples of the MNIST and Fashion-MNIST datasets, respectively.

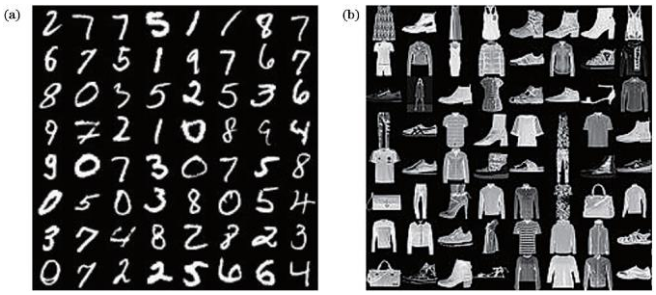

Figure 4. Some samples in MNIST and Fashion-MNIST datasets.
(a) MNIST dataset; (b) Fashion-MNIST dataset

4.2. Experiment settings

The experiment uses two standard unsupervised evaluation indicators to evaluate the clustering performance of the algorithm, namely ACC and NMI [27]. The two metrics have different characteristics in clustering tasks, and a higher value indicates better clustering performance. The experiment software environment is the Ubuntu16.04 system and the hardware environment is i7-6700 processor and NVIDIA GeForceGTX1060 graphics card, Python language, and the deep learning framework Pytorch [28,29]. In the experiment, the image is machine-disrupted within each batch, and a negative sample is selected in randomly disturbed order. In order to reduce the number of hyper-parameter searches, the nearest neighbor is set to k=3, and the transfer operator δ is 0.5. To reduce random errors, 10 experiments are performed under the same conditions, with an average of 10 experimental results. The number of channels and core size settings for the auto-encoding network are shown in table 2.

Table 2. Number of channels and core size of auto-encoder network

Dataset	Encoder-1/	Encoder-2/	Encoder-3/	Encoder-4/

	Decoder-4	Decoder-3	Decoder-2	Decoder-1
USPS	3×3×16	3×3×32		
MNIST	3×3×16	3×3×16	3×3×32	3×3×32
Fashion-MNIST	3×3×16	3×3×16	3×3×32	3×3×32

4.3. Experiment results

The ACC and NMI on the three data sets of MNIST, USPS, and Fashion-MNIST are shown in Table 3. As can be seen from Table 3, compared with other comparison algorithms, the proposed CCGCN algorithm has achieved the highest ACC and NMI values on all three classical image data sets. The proposed method can improve the clustering energy to achieve better experimental results. Especially, on the complex Fashion-MNIST dataset, the proposed algorithm still produces the best results. On the three graph sets of USPS, MNIST, and Fashion-MNIST, the CCGCN method is 1.97%, 2.16%, and 3.18% higher than the sub-optimal clustering method.

Table 3. Clustering results of different clustering algorithms on three datasets

Method	USPS		MNIST		Fashion-MNIST	
	ACC	NMI	ACC	NMI	ACC	NMI
K-means	0.6793	0.6381	0.5433	0.5115	0.4853	0.5231
AE+K-means	0.7042	0.6731	0.8187	0.7414	0.5964	0.6253
DEC	0.7519	0.7640	0.8766	0.8483	0.5291	0.5573
IDEC	0.7716	0.7957	0.8917	0.8783	0.5402	0.5681
Deepcluster	0.5734	0.5514	0.8082	0.6726	0.5533	0.5211
SDCN	0.7900	0.8037	0.8641	0.8538	0.5891	0.6158
Proposed	0.8097	0.8253	0.9133	0.9069	0.6282	0.6417

Depth-based clustering is generally better than traditional clustering methods, such as K-means algorithms. This is mainly because compared with the shallow clustering method, the deep neural network has the ability of dimensional reduction, which can effectively simulate the distribution of inputs, capture the nonlinear characteristics of inputs and learn well-learned deep-layer characteristics. Therefore, when dealing with high-dimensional nonlinear data, the clustering performance based on the depth clustering model is mostly better than that of the shallow clustering model. Compared with other methods based on depth clustering, such as AE, DEC, IDEC, and Deep-cluster, the SDCN algorithm achieves a better predictive effect by combining the sample's own attribute information and structure information to achieve a collaborative study of the sample's own attribute information and structural information. Compared with the SDCN algorithm, the proposed algorithm embeds the mutual information estimation network and minimizes the a priori distribution constraint in the multi-layer convolutional encoder, effectively excavates the deep characteristics of more identifiable samples, and makes the coding space more regular, and improves the coding quality of unsupervised feature extraction, which in turn improves clustering performance. Experimental results show that the new method achieves better clustering results than current advanced algorithms on three classical data sets.

4.4. Ablation experiment

A series of experiments are conducted on four different training strategies for the proposed model to verify the effectiveness of the CCGCN algorithm. 1) Train only one multi-layer convolutional encoder (ConvAE); 2) Embedding mutual information estimation network (ConvAE+MI) training model in convolutional encoder; 3) Convolution auto-coder and graph convolutional neural

network (ConvAE+GCN) to participate in model training; 4) Join the above three strategies (ConvAE+MI+GCN) to participate in model training [30-34]. These four training strategies have a slight effect on clustering, as shown in table 4.

Table 4. Impact of different strategies on clustering performance

Method	USPS		MNIST		Fashion-MNIST	
	ACC	NMI	ACC	NMI	ACC	NMI
ConvAE	0.7092	0.6630	0.7873	0.7561	0.5573	0.5674
ConvAE+MI	0.7964	0.7553	0.8461	0.8134	0.6033	0.6202
ConvAE+GCN	0.7933	0.7986	0.8685	0.8560	0.5954	0.6278
ConvAE+MI+GCN	0.8097	0.8253	0.9133	0.9069	0.6282	0.6417

Table 4 clearly shows that each training strategy can effectively improve the clustering performance on the basis of multi-layer convolutional auto-encoder, especially after adding the mutual information estimation network and the structure information of fused samples into ConvAE, the clustering effect is significantly improved. Since multi-layer convolution encoder in strategy (2) embedded in the mutual information to estimate the network. Since the global mutual information between input and potential feature representation and the local mutual information between mid-layer feature and potential feature representation are considered at the same time, especially the local mutual information is equivalent to treating each small part as a sample, so that the original sample becomes 1+M×M samples. It greatly increases the sample size and improves the coding quality of unsupervised feature extraction. In the USPS data set, ACC and NMI of strategy (2) are improved by 0.087 and 0.092 respectively compared with strategy (1). In MNIST data set, ACC and NMI are improved by 0.059 and 0.057, respectively. In the Fashion MNIST data set, ACC and NMI is improved by 0.046 and 0.053, respectively. In strategy (3), the features learned by different layers in ConvAE are integrated into the corresponding layers of GCN module, so that the model can simultaneously learn the attribute information of the sample itself and the structural information between the samples. So the strategy of combining ConvAE with GCN also produces better results than using ConvAE alone. In USPS data set, ACC and NMI is increased by 0.084 and 0.136 respectively compared with strategy (1). In MNIST data set, ACC and NMI increases by 0.081 and 0.100, respectively. In the Fashion MNIST data set, ACC and NMI increase by 0.038 and 0.060, respectively. In strategy (4), the above three strategies are combined to jointly optimize feature extraction and clustering allocation process end-to-end, and finally the model produces a stronger prediction effect. In the USPS data set, ACC and NMI of strategy (4) increase by 0.013 and 0.016 and 0.07 and 0.027, respectively, compared with strategy (2) and strategy (3). In MNIST data set, ACC and NMI are increased by 0.094 and 0.051, respectively. In the Fashion MNIST data set, ACC increases by 0.025 and 0. 033, and NMI increases by 0.022 and 0.014, respectively.

By using the t-SNE visualization method, clustering results for different training strategies are visualized in the MNIST dataset, as shown in figure 5. Figure 5(a) shows the distribution of data points in convAE's potential subspace, and Figure 5(b)~(d) shows the distribution of data points in subspace for the different strategies of the proposed model. From the distribution of potential space in figure 5, the data points in the embedded subspace obtained by model training that are jointly involved in the three strategies have a clearer distribution structure.

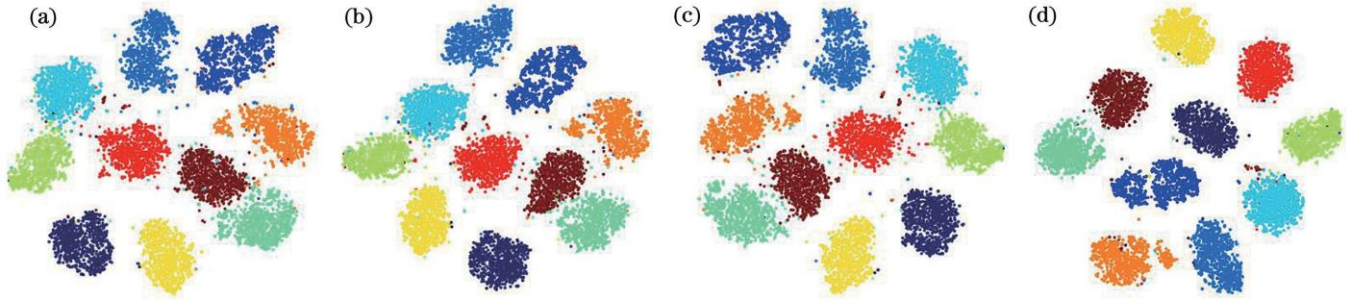

Figure 5. Distribution visualization of embedded subspaces of different strategies on the MNIST dataset. (a) ConvAE; (b) ConvAE+MI; (c) ConvAE+GCN; (d) ConvAE+ MI+GCN

4.5. Algorithm parameter evaluation experiment

To study the sensitivity of the CCGCN algorithm to parameters λ_1 and λ_2, ACC and NMI are used to assess the effects of different parameters on clustering energy. Figure 6 shows how the parameter λ_1 changes when searching in spaces {0.001,0.005,0.05,0.05,0.1}, and λ_2 in space {1,1.25,1.5,1.75,2}. As can be seen, the parameters λ_1 and λ_2 have varying degrees of impact on ACC and NMI in different combinations. But in most parameter combinations, ACC and NMI maintain relatively stable results.

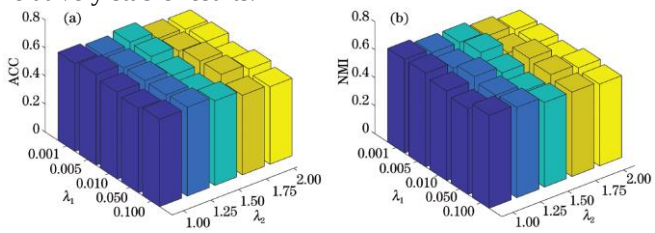

Figure 6. Effect of different combinations of parameters λ_1, and λ_2 on the ACC and NMI on the Fashion-MNIST dataset.(a) Effect on ACC; (b) Effect on NMI

5. Conclusion

In order to effectively improve the ability of deep feature identification, make full use of the structural information between unlabeled samples, jointly optimize the feature extraction and clustering process of samples, this paper proposes a CCGCN clustering algorithm. The algorithm embeds the mutual information estimation network and minimizes the a priori distribution constraint in the convolutional auto-coding network, and considers the property information of the imported sample itself and the structural information between the samples, which effectively improves the ability of feature discrimination while retaining more available structural information. On image clustering tasks, the CCGCN algorithm uses K-L diffuse joint to produce the potential feature distribution. Experimental results show that the clustering accuracy of CCGCN algorithm on three classical image data sets has been significantly improved. Especially on the complex Fashion-MNIST dataset, the accuracy of the proposed method is improved by 3.18% compared to the sub-optimal clustering algorithm. However, the effectiveness of the CCGCN algorithm is only validated on smaller image datasets, and how to effectively improve clustering performance on more large data sets is the focus of the next study.

Acknowledgements.

The authors thank the reviewers for their anonymous comments.

References

[1] Shoulin Yin, Ye Zhang, Shahid Karim. Large Scale Remote Sensing Image Segmentation Based on Fuzzy Region Competition and Gaussian Mixture Model[J]. IEEE Access. volume 6, pp: 26069 - 26080, 2018.

[2] Lin Teng, Hang Li, Shoulin Yin, Yang Sun. Improved krill group-based region growing algorithm for image segmentation[J]. International Journal of Image and Data Fusion. 10(4), pp. 327-341, 2019. doi: 10.1080/19479832.2019.1604574

[3] Shoulin Yin, Ye Zhang and Shahid Karim. Region search based on hybrid convolutional neural network in optical remote sensing images[J]. International Journal of Distributed Sensor Networks, Vol. 15, No. 5, 2019. DOI: 10.1177/1550147719852036

[4] Yang M, Deng C, Nie F. Adaptive-Weighting discriminative regression for multi-view classification[J]. Pattern Recognition, 2019, 88:236-245.

[5] Zeng Chaoping, Ju Lijun, Zhang Jianchen. Hyperspectral Image Classification Based on Clustering Dimensionality Reduction and Visual Attention Mechanism[J]. Laser & Optoelectronics Progress, 2019, 56(21):212802.

[6] Jing Yu, Hang Li, Shoulin Yin. New intelligent interface study based on K-means gaze tracking[J]. International Journal of Computational Science and Engineering, vol. 18, no. 1, pp. 12-20, 2019.

[7] Yang M, Tang J, Liu H, et al. A Novel Demodulation Method Based on Spectral Clustering for Phase-Modulated Signals Interrupted by the Plasma Sheath Channel[J]. IEEE Transactions on Plasma Science, 2020, 48(10):3544-3551.

[8] Shoulin Yin, Hang Li, Asif Ali Laghari, et al. A Bagging Strategy-Based Kernel Extreme Learning Machine for Complex Network Intrusion Detection[J]. EAI Endorsed Transactions on Scalable Information Systems. 21(33), e8, 2021. http://dx.doi.org/10.4108/eai.6-10-2021.171247

[9] Yu Z, Zhang Z, Cao W, et al. GAN-based Enhanced Deep Subspace Clustering Networks[J]. IEEE Transactions on Knowledge and Data Engineering, 2020, PP(99).

[10] Qingwu Shi, Shoulin Yin, Kun Wang, et al. Multichannel convolutional neural network-based fuzzy active contour model for medical image segmentation. Evolving Systems (2021). https://doi.org/10.1007/s12530-021-09392-3

[11] Ting-Ting Gao, Hang Li, and Shou-Lin Yin. Adaptive Convolutional Neural Network-based Information Fusion for Facial Expression Recognition [J]. International Journal of Electronics and Information Engineering. Vol. 13, No. 1, pp. 17-23, 2021.

[12] Yang Sun, Shoulin Yin, and Lin Teng. Research on Multi-robot Intelligent Fusion Technology Based on Multi-mode Deep Learning [J]. International Journal of Electronics and Information Engineering. Vol. 12, No. 3, pp. 119-127, 2020.

[13] Xie J, Girshick R, Farhadi A. Unsupervised Deep Embedding for Clustering Analysis[J]. Computer Science, 2015.

[14] Bashon Y, Neagu D, Ridley M J. A framework for comparing heterogeneous objects: on the similarity measurements for fuzzy, numerical and categorical attributes[J]. Soft Computing, 2013, 17(9):1595-1615.

[15] Caron M, Bojanowski P, Joulin A, et al. Deep Clustering for Unsupervised Learning of Visual Features[C]// European Conference on Computer Vision. Springer, Cham, 2018.

[16] Shahid Karim, Ye Zhang, Shoulin Yin, Irfana Bibi. A Brief Review and Challenges of Object Detection in Optical Remote Sensing Imagery [J]. Multiagent and Grid Systems. 16(3), 227-243, 2020

[17] S. Yin and H. Li. Hot Region Selection Based on Selective Search and Modified Fuzzy C-Means in Remote Sensing Images[J]. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 5862-5871, 2020, doi: 10.1109/JSTARS.2020.3025582.

[18] Zhan X, Xie J, Liu Z, et al. Online Deep Clustering for Unsupervised Representation Learning[C]// 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2020.

[19] Chen M, Chang Z, Lu H, et al. AugNet: End-to-End Unsupervised Visual Representation Learning with Image Augmentation[J]. 2021. arXiv:2106.06250

[20] Hjelm R D, Fedorov A, Lavoie-Marchildon S, et al. Learning deep representations by mutual information estimation and maximization[J]. 2018. arXiv:1808.06670

[21] A. Abdella and I. Uysal, "A Statistical Comparative Study on Image Reconstruction and Clustering With Novel VAE Cost Function," in IEEE Access, vol. 8, pp. 25626-25637, 2020, doi: 10.1109/ACCESS.2020.2971270.

[22] Kip F T N, Welling M. Semi-Supervised Classification with Graph Convolutional Networks[J]. 2016. arXiv:1609.02907

[23] Kipf T N, Welling M. Variational Graph Auto-Encoders[J]. 2016. arXiv:1611.07308

[24] Yin, S., Li, H. & Teng, L. Airport Detection Based on Improved Faster RCNN in Large Scale Remote Sensing Images [J]. Sensing and Imaging, vol. 21, 2020. https://doi.org/10.1007/s11220-020-00314-2

[25] Ying Z, You J, Morris C, et al. Hierarchical graph representation learning with differentiable pooling [J]. Neural Inform Process Systems, 2018, 9: 4805-15.

[26] Xingwu Fu, Mingming Lv, Wanjun Liu, et al. Structured Deep Discriminant Embedded Coding Network for Image Clustering [J]. Laser & Optoelectronics Progress, 58(6), 2021.

[27] Shoulin Yin, Ye Zhang and Shahid Karim. Region search based on hybrid convolutional neural network in optical remote sensing images[J]. International Journal of Distributed Sensor Networks, Vol. 15, No. 5, 2019. (JA) DOI: 10.1177/1550147719852036

[28] Lin Teng, Hang Li, Shoulin Yin, Yang Sun. Improved krill group-based region growing algorithm for image segmentation[J]. International Journal of Image and Data Fusion. 10(4), pp. 327-341, 2019. doi: 10.1080/19479832.2019.1604574

[29] Shoulin Yin, Lei Meng and Jie Liu. A New Apple Segmentation and Recognition Method Based on Modified Fuzzy C-means and Hough Transform[J]. Journal of Applied Science and Engineering. Vol. 22, No. 2, pp. 349-354, 2019.

[30] Laghari, A.A., Wu, K., Laghari, R.A. et al. A Review and State of Art of Internet of Things (IoT). Arch Computat Methods Eng (2021). https://doi.org/10.1007/s11831-021-09622-6

[31] Laghari A A, Laghari M A. Quality of experience assessment of calling services in social network[J]. ICT Express, 2021(2).

[32] Laghari A A, Laghari K, Memon K A, et al. Quality of Experience (QoE) Assessment of Games on workstations and Mobile[J]. Entertainment Computing, 2020, 34:100362.

[33] A. A. Laghari, H. He, A. Khan, N. Kumar and R. Kharel, "Quality of Experience Framework for Cloud Computing (QoC)," in IEEE Access, vol. 6, pp. 64876-64890, 2018, doi: 10.1109/ACCESS.2018.2865967.

[34] Laghari, A.A., Jumani, A.K. & Laghari, R.A. Review and State of Art of Fog Computing. Arch Computat Methods Eng 28, 3631–3643 (2021). https://doi.org/10.1007/s11831-020-09517-y