International Journal of Machine Learning and Cybernetics
https://doi.org/10.1007/s13042-022-01533-7

ORIGINAL ARTICLE

Unsupervised image clustering algorithm based on contrastive learning and K-nearest neighbors

Xiuling Zhang[1,2] · Shuo Wang[1] · Ziyun Wu[1] · Xiaofei Tan[1]

Received: 23 June 2021 / Accepted: 22 February 2022

Abstract
With the development of the times, people generate a huge amount of data every day, most of which are unlabeled data, but manual labeling needs a lot of time and effort, so unsupervised algorithms are being used more often. This paper proposes an unsupervised image clustering algorithm based on contrastive learning and K-nearest neighbors (CLKNN). CLKNN is trained in two steps, which are the representation learning step and the clustering step. Contrastive learning and K-nearest neighbors have a huge impact on CLKNN. In the representation learning step, firstly CLKNN processes the image by double data augmentation to get two different augmented images; then CLKNN uses double contrastive loss to extract the high-level feature information of the augmented images, maximizing the similarity of row space and maximizing the similarity of column space to ensure the invariance of information. In the clustering step, CLKNN finds the nearest neighbors of each image by K-nearest neighbors, then it maximizes the similarity between each image and its nearest neighbors to get the final result. To test the performance of CLKNN, the experiments are conducted on CIFAR-10, CIFAR-100 and STL-10 in this paper. From the final results, it is clear that CLKNN has better performance than other advanced algorithms.

Keywords Contrastive learning · K-nearest neighbors · Double data augmentation · Double contrastive loss

1 Introduction

Today, deep learning has been widely used in many domains, including speech recognition, image segmentation, and natural language processing. However, many current supervised deep learning tasks tend to rely excessively on manual labeling, which is usually costly to collect data manually and also difficult to scale the dataset, and this becomes some disadvantages of supervised learning. Since there are huge quantities of unlabeled data in real-world problems, forcing the object of many studies to shift from a large amount of data with labels to data with only a small amount of labels or no labels, thus the research methods have changed from supervised learning to unsupervised learning. In a word, unsupervised learning for image tasks has a more important significance in reality. The proposal of self-supervised learning breaks the limitation of manually labeled samples and aims to efficiently train the network for unsupervised semantic extraction even without manual labeling. The core problem of self-supervised learning is how to generate pseudo labels, for example, the pseudo label of a common autoencoder (AE) is the original image itself. In practice, self-supervised learning has many achievements, such as denoising autoencoder, contextual autoencoder, or cross-channel autoencoder, and we hope that the intermediate feature representations can contain more semantic information and can be used for various downstream tasks. Unsupervised learning has many tasks in various fields, such as image classification [1], image clustering [2–4], image segmentation [5], target detection [6], etc. Although the tasks are different, they all aim to extract more useful features and simplify the subsequent operations to better complete the tasks. In the image clustering task, it is especially important to extract features that reduce the distance within the same class and increase the distance between dissimilar classes.

✉ Shuo Wang
352051330@qq.com

Xiuling Zhang
zxlysu@ysu.edu.cn

[1] Engineering Research Center of the Ministry of Education for Intelligent Control System and Intelligent Equipment, Yanshan University, Qinhuangdao 066004, China

[2] Key Laboratory of Industrial Computer Control Engineering of Hebei Province, Yanshan University, Qinhuangdao 066004, China

Published online: 08 March 2022

The earliest ideas of unsupervised clustering were focused on K-means, DBSCAN algorithm, while later clustering studies used restricted Boltzmann machines [7] and autoencoder [8] to extract features. LeNet-5 [9] and discriminative unsupervised algorithm [10] verifies the feasibility of convolutional neural networks for unsupervised learning. Deep convolutional neural networks have powerful capabilities in image feature extraction [11], and convolutional autoencoder [12] provides a new way of image feature extraction. Generative adversarial networks [13] are also used for feature extraction. An unsupervised algorithm based on adversarial networks [14] proves its feasibility. The unsupervised clustering methods [2] combine the ideas of clustering and classification to learn visual features. Up to this point, unsupervised clustering learning has evolved from stepwise training of features and clusters to joint training.

Many current supervised deep learning tasks tend to rely excessively on manual labeling, which is usually costly to collect data manually and also difficult to expand the datasets. However, unsupervised algorithms can avoid the problem of manually labeled data. Unsupervised algorithms use unlabeled data to train the network, perform unsupervised semantic extraction with unlabeled data, and finally achieve some tasks such as image clustering, image segmentation, and target detection. But, the current image clustering algorithm has some shortcomings, such as the feature extraction method is not advanced, the processing of feature information is not reasonable enough, and the accuracy of the model is low.

In order to improve these shortcomings of the image clustering algorithm, an unsupervised image clustering algorithm named as unsupervised image clustering algorithm based on contrastive learning and K-nearest neighbors (CLKNN) is proposed. Unlike the recently popular algorithm of joint training, CLKNN is a two-step algorithm. The first step is the representation learning step, CLKNN trains the model by using a pretext task, in which the input image is processed by double data augmentation, then CLKNN performs feature extraction based on double contrastive loss function. The second step is the clustering step, CLKNN obtains the final result by maximizing the similarity between each sample and its nearest neighbors.

The main contributions are summarized in 4 major points:

1. CLKNN is an unsupervised clustering algorithm. CLKNN uses double data augmentation to process each image and obtain two different augmented images. The augmented images are used for representation learning, which can expand samples, prevent overfitting, and improve robustness.
2. CLKNN proposes double contrastive loss to improve the effectiveness of representation learning. In the feature space, the number of rows represents the number of samples and the number of columns represents the dimensionality of the features. When the number of columns is equal to the number of image classes, a row of data stands for the probability that a certain image is assigned to each class, and a column of data stands for the probability that each sample is assigned to this class. Double contrastive loss ensures information invariance by maximizing the similarity of row space and the similarity of column space.
3. In the clustering step, CLKNN uses K-nearest neighbors to finds the nearest neighbors of each image and then maximizes the similarity between each image and its nearest neighbors to obtain the final result.
4. In this paper, all experiments are conducted on three datasets, and the experimental result indicates that CLKNN possesses better performance than other unsupervised clustering algorithms.

2 Related work

With the help of advanced theories, unsupervised learning started to use neural networks to improve algorithm and it has been developed rapidly. The convolutional autoencoders and contrastive learning used in supervised learning are also applied in unsupervised learning.

2.1 Deep clustering

Traditional clustering algorithms have achieved good results on large and complex datasets, but the lack of representation learning capability makes the clustering results disappointing. Previous unsupervised clustering algorithms are independent of each other in feature extraction and clustering, how to unite the training of deep convolutional networks with the clustering becomes an important direction for unsupervised clustering learning. Deep clustering gets excellent results on many datasets after adding the representation learning. In 2016, Xie et al. [15] proposed the DEC algorithm that jointly solves feature learning and clustering relationship discrimination based on convolutional autoencoders. DEC also has some improvements at the level of generating features. The cluster centers of the features are generated by initializing the data-to-feature mapping with deep autoencoders. It uses t-distribution to measure the similarity between feature points and feature cluster centers, then constructs the auxiliary target distribution by generating the clustering probability distribution of each data in a soft assignment way. The KL-divergence is used to optimize the cluster centers, while the parameters of the generated feature network are continuously adjusted to optimize the clustering results until the network converges. The accuracy of the final clustering results surpasses previous clustering

methods. The DEC algorithm is a major advancement in unsupervised learning, because feature learning and clustering discrimination are no longer independent of each other, they can be jointly trained and optimized together. However, the deep autoencoders still have limitations in the extraction of features.

To optimize this problem, DEPICT [16] was proposed. DEPICT uses a multinomial logistic regression function to calculate the similarity of feature points to the cluster centers, while keeping the overall idea of the original unsupervised deep embedding clustering algorithm; the network is also modified to a deep convolutional denoising autoencoder; the input data without noise is used to generate the auxiliary target distribution, and then DEPICT optimizes clustering probability distribution for the input data with noise. DBC [17] optimized deep autoencoder to a deep convolutional autoencoder and improved the accuracy of cluster assignment by discriminatively boosted assignment, the better results were obtained with other aspects unchanged.

JULE [18] achieves end-to-end learning through joint representation learning and clustering. Similarly, Deep Clustering [19] uses k-means to group features and updates the deep network based on cluster assignment. Although this approach can learn representation and perform clustering simultaneously, the errors accumulated may affect their performance during the alternation. In addition, it requires clustering the entire dataset, which affects the scope of its application. PICA [20] separates a group of images by maximizing the partition confidence of clustering solution. IIC [21] achieves clustering by maximizing the mutual information of data pairs. These work, while having a theoretical basis, relies heavily on auxiliary overclustering techniques which is difficult to interpret.

Unlike the end-to-end learning approach of current image clustering algorithms, the CLKNN algorithm is a two-step algorithm. CLKNN divides the training process into representation learning step and clustering step. In the representation learning step, the CLKNN algorithm uses double data augmentation to generate double samples while generating pseudo-labels. Then CLKNN uses a double contrastive loss function to update the neural network according to the pseudo-labels. In the clustering step, CLKNN uses K-nearest neighbor loss to maximize the similarity between each sample and its neighboring samples to further enhance the learning ability of the neural network. These innovations make the CLKNN algorithm better than the most image clustering algorithms.

2.2 Contrastive learning

Contrastive learning is also widely used in deep learning. Contrastive learning is to map the input data into a feature space, which divides the samples into positive and negative pairs according to whether the samples are the same class or not, then it designs a loss function to maximize the similarity of the positive pairs and minimize the similarity of the negative pairs. Contrastive learning is commonly used in the fields of face recognition, pedestrian re-identification and image clustering.

As early as 2006, Le Cun et al. [22] proposed contrastive loss, which has the important role of making similar samples as close as possible and dissimilar samples as distant as possible from each other during the training process. However, this original contrastive loss requires to know the label of each sample and judge whether the two samples are similar. Several studies in recent years have proposed improved contrastive loss, such as Noise-Contrastive Estimation (NCE), Info NCE, etc, which are popular in the unsupervised field of contrastive learning. NCE was first proposed in 2012 as a new approach to statistical model, and it has been successful in natural language processing and image classification. In subsequent studies, NCE has also been widely applied. For example, Mnih et al. [23] introduced noise contrast evaluation into neural network language models to improve the performance of the model. Li et al. [24] proposed a general unsupervised learning method named as contrastive predictive coding that can extract useful representations from high-dimensional data, and its model uses Info NCE as a constraint, and this loss can capture the most useful information for predicting unknown samples in the potential space. He et al. [25] proposed a contrastive method (MoCo), which constructs a special dictionary with a momentum encoder and a queue from the perspective of contrastive learning. Therefore, MoCo is able to quickly construct a large and consistent dictionary to facilitate contrastive learning. Henaff et al. [26] improved contrastive predictive coding and applied it to the task of image recognition, which greatly improves the performance of target detection and recognition over the fully supervised pre-trained classifier on ImageNet. Chen et al. [27] proposed a simple framework for contrastive learning of visual representations (SimCLR), where data augmentation defines predictive tasks.

3 Proposed CLKNN algorithm

CLKNN is an unsupervised algorithm based on neural networks. CLKNN uses contrastive learning to extract high-level feature of images. Then CLKNN uses K-nearest neighbors to find the nearest neighbors of each image. Finally, CLKNN maximizes the similarity between each image and its nearest neighbors to get the final result.

3.1 Pretext task

In order to avoid manually labeled samples in deep learning and achieve unsupervised semantic extraction, the pretext task is a good choice, which has the advantage of simplifying the solution of the original task. Pretext task can be further understood as an auxiliary task that is helpful to the target task, and it is now often used for self-supervised learning, also known as a more general type of unsupervised learning, it breaks this limitation of manually labeled samples and aims to train the network efficiently without manually labeled conditions. Usually, the pretext task also involves pre-training and fine-tuning, where pre-training refers to a pre-trained model or the process of pre-training a model. In other words, some initialized parameters have been obtained in advance, which is not random but learned through other similar datasets, and then the model is trained with original dataset to get the parameters suitable for original dataset; Fine-tuning is the process of applying a pre-trained model to original dataset and adapting the parameters to original dataset. Therefore, this paper applies the pretext task to embedded clustering networks. On the one hand, a pre-trained model can be used to initialize the embedded clustering network with parameters learned in other complex datasets and has a better generalization capability; on the other hand, fine-tuning can be used to simplify the process of unsupervised embedded clustering to learn features.

3.2 Framework for CLKNN

In representation learning, CLKNN gets the representation learning function $\boldsymbol{\Phi}_{\Theta}$ by the pretext task, where Θ denotes the parameters in ResNet, and the image is mapped to feature space by this function. The framework of CLKNN is shown in Fig. 1. First, CLKNN augments the input image x with data augmentation methods T^1 and T^2 to obtain the augmented data x^1 and x^2. CLKNN extracts the feature representation of x^1 and x^2 by $\boldsymbol{\Phi}_{\Theta}$, the feature representation of the images extracted from each batch of size are divided into y^1 and y^2 according to the different data augmentation methods. I^1 and I^2 are obtained from y^1 and y^2 by f_I, f_I is the MLP used for the instance-level contrastive part; C^1 and C^2 are obtained from y^1 and y^2 by f_C, f_C is the MLP used for the cluster-level contrastive part; CLKNN performs contrastive learning in row space of I^1 and I^2; CLKNN performs contrastive learning in column space of C^1 and C^2.

Next, several innovations of the CLKNN algorithm are discussed in detail.

1. Double data augmentation. The pretext task is an auxiliary task that is helpful to a target task, and double data augmentation is a way to achieve the pretext task. Double data augmentation consists of ColorJitter, GaussianBlur, Grayscale, HorizontalFlip, and ResizedCrop. The target task of image clustering algorithm is to complete the classification of unlabeled images, but it is difficult to get accurate classification results because there is no label information. Therefore the CLKNN sets up a auxiliary task to assist in the target task. In the auxiliary task, CLKNN uses double data augmentation to obtain the augmented images and get pseudo-labels. With the help of pseudo-labels, CLKNN is trained with unlabeled samples and achieves unsupervised semantic extraction.
2. Double contrastive loss. The idea of contrastive loss is to map the original data into the feature space in which the similarity of positive pairs is maximized and the similarity of negative pairs is minimized. The positive pair consists of two augmented images from the same image, and the negative pair otherwise. Contrastive loss calculates the similarity in row space only at the instance level, which can improve the representation learning ability of the model to some extent, but there is potential for improvement. Therefore, double contrastive loss is proposed in this paper. In addition to calculating the similarity in row space at the instance level, the double contrastive loss also calculates the similarity in column space at the cluster level. The double contrastive loss combines feature information in both row space and col-

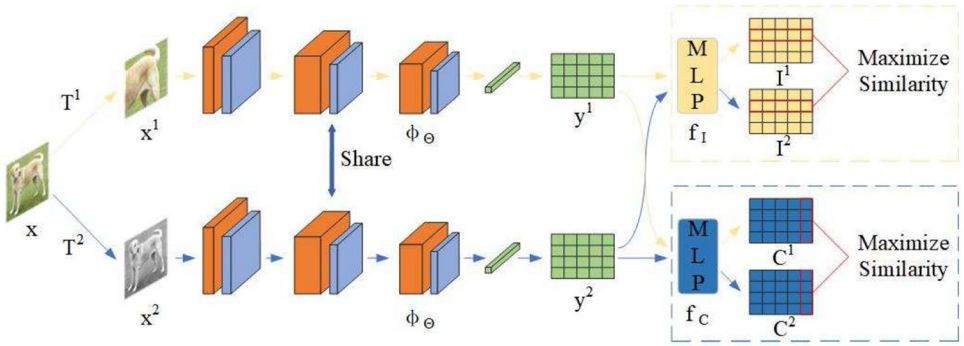

Fig. 1 The framework of CLKNN

umn space, which can greatly improve the representation learning ability of the model.

3. K-nearest neighbor loss. If the image clustering algorithm simply applies K-means to the obtained features in the clustering step, the clustering effect is not good. However, the idea of K-nearest neighbors can be used to improve the clustering effect. First, the CLKNN algorithm finds the K nearest neighbors of each sample based on the feature information. The nearest neighbors of some samples are shown in Fig. 2. As can be seen from Fig. 2, it is high probability that each sample and its nearest neighbors belong to the same class. Therefore, K-nearest neighbor loss is to further optimize the model by maximizing the similarity between each image and its K most similar images.

3.3 Representation learning for CLKNN

First, this section describes the calculation process of contrastive loss in the representation learning step. The data augmentation methods T^1 and T^2 are the same as in SimCLR [27], but CLKNN uses two times data augmentation for the same image and gets two different augmented images, which is named as double data augmentation in this paper.

CLKNN augments the input image x_i with data augmentation methods T^1 and T^2, and gets 2N samples $\{x_1^1, x_2^1, \ldots, x_N^1; x_1^2, x_2^2, \ldots, x_N^2\}$. Unlike the way positive and negative pairs are defined in supervised learning, CLKNN is an unsupervised learning method. Instead, this paper defines positive pairs as two augmented images from the same image and negative pairs as two augmented images from the different images. For the image x_i, the positive pair is composed of $\{x_i^1, x_i^2\}$ and the negative pair is composed of the other 2N-2 pairs.

Fig. 2 The nearest neighbors of each some samples

To reduce information loss, CLKNN does not perform contrastive learning on the feature matrix. Instead, CLKNN constructed two two-layer MLP including f_I and f_C, CLKNN maps the feature matrix $\{y^1, y^2\}_{N\times M}$ to a smaller space by f_I and f_C, then CLKNN obtain $\{I^1, I^2, C^1, C^2\}$. Finally, CLKNN performs contrastive loss in $\{I^1, I^2, C^1, C^2\}$ and the similarity is defined by cosine distance. The similarity is defined as follows:

$$\mathrm{sim}\left(I_i^1, I_j^2\right) = \frac{\left(I_i^1\right)\left(I_j^2\right)^T}{\left\|\left(I_i^1\right)\right\|\left\|\left(I_j^2\right)\right\|} \tag{1}$$

where $i, j \in \{1, 2, \ldots, N\}$. In order to maximize the similarity of positive pairs and minimize the similarity of negative pairs, this paper takes x_i as an example and calculates its loss at the instance level. The loss of x_i is calculated as follows:

$$L_i^s = -\log \frac{\exp\left(\mathrm{sim}\left(I_i^1, I_i^2\right)/\tau\right)}{\sum_{j=1}^{N}\left[\exp\left(\mathrm{sim}\left(I_i^s, I_j^s\right)/\tau\right) + \exp\left(\mathrm{sim}\left(I_i^1, I_j^2\right)/\tau\right)\right]} \tag{2}$$

where τ is the temperature coefficient, $s \in \{1, 2\}$. CLKNN needs to calculate the instance-level contrastive loss for all augmented samples, so the sum of the losses at the instance level is as follows:

$$L_I = \frac{1}{2N}\sum_{i=1}^{N}\left(L_i^1 + L_i^2\right) \tag{3}$$

The process of calculating cluster-level losses is similar to that of instance-level losses, so we give the calculation formula as follows:

$$L_i'^s = -\log \frac{\exp\left(\mathrm{sim}\left(C_i^1, C_i^2\right)/\tau\right)}{\sum_{j=1}^{M}\left[\exp\left(\mathrm{sim}\left(C_i^s, C_j^s\right)/\tau\right) + \exp\left(\mathrm{sim}\left(C_i^1, C_j^2\right)/\tau\right)\right]} \tag{4}$$

CLKNN needs to calculate the cluster-level contrastive loss for all augmented samples, so the sum of the losses at the cluster level is as follows:

$$L_C = \frac{1}{2M}\sum_{i=1}^{M}\left(L_i'^1 + L_i'^2\right) \tag{5}$$

In the representation learning step, the total loss is defined as:

$$L_{rep} = L_I + L_C \tag{6}$$

3.4 Clustering for CLKNN

In the clustering step, CLKNN defines a K-nearest neighbor loss. After CLKNN calculates the K-nearest neighbors of each sample, it maximizes the similarity between the sample and its nearest neighbors as part of the total loss to continuously optimize the clustering results. Here, this paper defines D as the set of original samples and $N(x_i)$ as the set of K-nearest neighbor of image x_i. $\boldsymbol{\Phi}_\beta$ is a clustering function and β is a parameter of the neural network. $\boldsymbol{\Phi}_\beta$ finally divides the image x_i into C classes by soft assignment, and $\phi_\beta(x_i)$ is a C-dimensional row vector. The K-nearest neighbor loss is calculated as follows:

$$L_K = -\frac{1}{|D|}\sum_{x_i \in D}\sum_{k \in N(x_i)} \log\left(\phi_\beta(x_i)\cdot\phi_\beta(k)^T\right) \tag{7}$$

To avoid assigning most samples to the same cluster, CLKNN sets an information entropy to solve this problem. The information entropy is calculated as follows:

$$L_H = -\sum_{i=1}^{M}\left[P\left(C_i^1\right)\log P\left(C_i^1\right) + P\left(C_i^2\right)\log P\left(C_i^2\right)\right] \tag{8}$$

where $P\left(C_i^s\right) = \sum_{j=1}^{N}\frac{C_{ij}^s}{\|C\|_1}$, $s \in \{1,2\}$. Finally, the model is constantly optimized by continuously minimizing the following losses:

$$L_{clu} = L_K + L_H \tag{9}$$

The training process of CLKNN is shown in Algorithm 1.

Table 1 Datasets overview

Dataset	Classess	Samples	Train	Test	Aspect ratio
STL-10	10	13,000	5000	8000	96×96
CIFAR-10	10	60,000	50,000	10,000	32×32
CIFAR-100	20	60,000	50,000	10,000	32×32

Table 2 Parameter setting of the datasets

Dataset	Pretext-batch of size	Pretext-epoch	Clustering-batch of size	Clustering-epoch
STL-10	64	500	128	100
CIFAR-10	128	500	512	100
CIFAR-100	128	500	512	100

Algorithm 1 CLKNN

Input: Dataset D; Temperature coefficient: τ; Number of nearest neighbors: K; Number of classes: C; Neural Networks: Φ_Θ, Φ_β; epoch: E_1, E_2.
Output: Low-dimensional feature matrix: I^1, I^2, C^1, C^2; Final results:$\phi_\beta(D)$.

```
Use the pretext task continuously to optimize Φ_Θ.
for epoch=1 to E_1 do
  for x_i ∈ {D} do
    Augment the input image x_i with double data augmentation to get x_i^1 and x_i^2.
    x_i^1 and x_i^2 are trained by the neural network Φ_Θ to get the feature matrices y_i^1 and y_i^2.
    y_i^1 and y_i^2 are mapped by f_I and f_C to get I^1, I^2, C^1, C^2.
    Update Φ_Θ with the loss function L_rep.
    Compute the nearest neighbors of sample x_i to obtain N(x_i).
  end for
end for
for epoch=1 to E_2 do
  Update Φ_β with loss function L_clu
end for
Return Final results:φ_β(D).
```

Table 3 Ablation experiments on CIFAR-10

Setup	ACC	ARI	NMI
Orgin	78.8	61.5	66.7
Double data augmentation	79.3	62.2	67.0
Double contrastive loss	79.2	62.3	66.9
CLKNN	80.3	63.8	68.6

4 Experiments and analysis

The experimental datasets include STL-10, CIFAR-10 and CIFAR-100 in this paper, and the details of these datasets are shown in Table 1. Some recent algorithms are also trained and evaluated on these datasets, and these datasets are divided into train and test sets, the models of these algorithms are trained on the train sets, and then tested on the test sets. The results on the test sets can represent the generalization ability of the models, so the generalization ability of the models in these algorithms can be observed through the results on the test sets. In this paper, the same network, hyperparameters, and data augmentation are used to complete all experiments.

4.1 Experimental setup

CLKNN uses ResNet18 as the main network and finds 30 nearest neighbors by using NCE, and double data augmentation is used for the pretext task. First, CLKNN augments the input image by using double data enhancement; then, CLKNN uses the pretext task to train the model and transfers the weights from the pre-trained model to the clustering step; finally, CLKNN selects the model with the lowest loss based on the results of each iteration. This paper takes STL-10 as an example and introduces the parameter setting of CLKNN in the experiment. In the pretext task, this paper runs 500 iterations at the batch of size 64. In the clustering stage, this paper runs 100 iterations at the batch of size 128. The batch of size and number of iterations are shown in Table 2.

4.2 Ablation studies

It is well known that data augmentation has a significant impact on contrastive learning and different loss functions have a significant effect on the training results of the model. To test the effect of some improvements on CLKNN, ablation experiments are performed on CIFAR-10 in this paper. Table 3 shows the results of the ablation experiments, the original algorithm (SCAN)[28] with single data augmentation and single contrastive loss has the lowest accuracy (78.8%); the accuracy of the improved algorithm is 79.3% after improving only single data augmentation to double data augmentation, achieving an improvement of 0.5% compared to SCAN; The accuracy of the improved algorithm is 79.2% after improving only the single contrastive loss to double contrastive loss, achieving an improvement

Table 4 The clustering performance on three image datasets

Algorithm	CIFAR-10			CIFAR-100			STL-10		
Metrics	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI
K-means	22.9	4.9	8.7	13.0	2.8	8.4	19.2	6.1	12.5
SC	24.7	8.5	10.3	13.6	2.2	9.0	15.9	4.8	9.8
AC	22.8	6.5	10.5	13.8	3.4	9.8	33.2	14.0	23.9
NMF	19.0	3.4	8.1	11.8	2.6	7.9	18.0	4.6	9.6
AE	31.4	16.9	23.9	16.5	4.8	10.0	30.3	16.1	25.0
DAE	29.7	16.3	25.1	15.1	4.6	11.1	30.2	15.2	22.4
DCGAN	31.5	17.6	26.5	15.1	4.5	12.0	29.8	13.9	21.0
DeCNN	28.2	17.4	24.0	13.3	3.8	9.2	29.9	16.2	22.7
VAE	29.1	16.7	24.5	15.2	4.0	10.8	28.2	14.6	20.0
JULE	27.2	13.8	19.2	13.7	3.3	10.3	27.7	16.4	18.2
DEC	30.1	16.1	25.7	18.5	5.0	13.6	35.9	18.6	27.6
DAC	52.2	30.6	39.6	23.8	8.8	18.5	47.0	25.7	36.6
ADC	32.5	–	–	16.0	–	–	53.0	–	–
DDC	52.4	32.9	42.4	–	–	–	48.9	26.7	37.1
DCCM	62.3	40.8	49.6	32.7	17.3	28.5	48.2	26.2	37.6
IIC	61.7	–	–	25.7	–	–	61.0	–	–
PICA	69.6	51.2	59.1	33.7	17.1	31.0	71.3	53.1	61.1
SCAN	78.8	61.5	66.7	38.8	23.9	39.9	67.1	49.6	57.4
CLKNN	80.3	63.8	68.6	40.7	25.1	41.8	71.6	51.9	59.6

of 0.4% compared to SCAN; the CLKNN algorithm uses both improvements, the accuracy of CLKNN on CIFAR-10 is 80.3%, which is a 1.5% improvement over SCAN, a 1.0% improvement over double data augmentation, and a 1.1% improvement over double contrastive loss. According to the results of the ablation experiments, the improvement of SCAN in this paper is effective, and CLKNN has made a great improvement compared to SCAN.

4.3 Experimental result

To test the performance of CLKNN, three common evaluation metrics are used in the experiments, including Accuracy (ACC), and Adjusted Rand Index (ARI), Normalized Mutual Information (NMI). All the experimental results are presented in Table 4. In Table 4, the experiments of CLKNN and SCAN are performed on GTX 1080Ti, and the experimental results of other algorithms are derived from PICA [20] since the source code is not available. CLKNN outperforms the most algorithms in Table 4. CLKNN has huge advantages over the original clustering algorithms K-means and SC, CLKNN surpasses K-means by 57.4% on CIFAR-10. Compared with the original image clustering algorithm DEC, CLKNN surpasses DEC by 50.2% on CIFAR-10. Compared with the recent algorithm PICA, CLKNN surpasses PICA by 10.7% on CIFAR-10. Compared to the latest algorithm SCAN, CLKNN surpasses SCAN by 1.5% on CIFAR-10. The experimental results illustrate that CLKNN has powerful image clustering capabilities and outperforms most recent work.

CLKNN is almost the best algorithm on these datasets, but the results of PICA on STL-10 are better than those of CLKNN. PICA is lower than CLKNN by 0.3% on STL-10 in terms of ACC, PICA surpasses CLKNN by 1.2% in terms of ARI, PICA surpasses CLKNN by 1.5% in terms of NMI. This may be due to the small sample size of the STL-10 and the fact that the samples of train set are less than those of test set in STL-10. The small datasets are easy to make the model over fit and reduce the stability of the model.

5 Conclusion

This paper proposes an unsupervised image clustering algorithm based on contrastive learning and K-nearest neighbors (CLKNN). CLKNN augments the input image by using double data augmentation and then performs feature extraction based on the double contrastive loss. Double data augmentation can expand samples, prevent overfitting, and improve model robustness; double contrastive loss can ensure the invariance of information. Finally, CLKNN obtains the final result by maximizing the similarity between each sample and its nearest neighbors. In this paper, all experiments are conducted on three datasets, and the experimental result indicates that CLKNN possesses better performance than other unsupervised clustering algorithms.

Acknowledgements This work is supported by Hebei Provincial Department of education in 2021 provincial postgraduate demonstration course project construction under Grant KCJSX2021024.

References

1. Kang G, Jiang L, Yang Y, Hauptmann AG (2019) Contrastive adaptation network for unsupervised domain adaptation. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp 4888–4897
2. Caron M, Bojanowski P, Joulin A, Douze M (2018) Deep clustering for unsupervised learning of visual features. Lect Notes Comput Sci 20:139–156
3. Zeng S, Zhang B, Zhang Y, Gou J (2020) Dual sparse learning via data augmentation for robust facial image classification. Int J Mach Learn Cybernet 11:1717–1734
4. Chen Z, Ding S, Hou H (2021) A novel self-attention deep subspace clustering. Int J Mach Learn Cybernet. https://doi.org/10.1007/s13042-021-01318-4
5. Wang W, Song H, Zhao S et al (2019) Learning unsupervised video object segmentation through visual attention. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp 3059–3069
6. Vo HV, Bach F, Cho M et al (2019) Unsupervised image matching and object discovery as optimization. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp 8279–8288
7. Fischer A, Igel C (2012) An introduction to restricted Boltzmann machines. Lect Notes Comput Sci 20:14–36
8. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323(6088):533–536
9. Lecun Y, Bottou L, Bengio Y et al (1988) Gradient-based learning applied to document recognition. Proc IEEE Int Conf Comput Vis 86(11):533–536
10. Dosovitskiy A, Springenberg JT, Riedmiller M, Brox T (2014) Discriminative unsupervised feature learning with convolutional neural networks. Adv Neural Inf Process Syst 20:766–774
11. Hinton GE, Salakhutdinov RR (2006) Reducing the dimensionality of data with neural networks. Science 313(5786):504–507
12. Masci J, Meier U, Cireşan D et al (2011) Stacked convolutional auto-encoders for hierarchical feature extraction. In: The 21th international conference on artificial neural networks, pp 52–59
13. Goodfellow I, Pouget-Abadie J, Mirza M et al (2014) Generative adversarial nets. arXiv:1406.2601v1
14. Radford A, Metz L, Chintala S (2016) Unsupervised representation learning with deep convolutional generative adversarial networks. arxiv: 1511.06434
15. Xie J, Girshick R, Farhadi A (2016) Unsupervised deep embedding for clustering analysis. In: 33rd international conference on machine learning, vol 1, pp 740–749
16. Dizaji KG, Herandi A, Deng C et al (2017) Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization. In: Proceedings of the IEEE international conference on computer vision, pp 5747–5756
17. Li F, Qiao H, Zhang B (2018) Discriminatively boosted image clustering with fully convolutional auto-encoders. Pattern Recogn 83:161–173
18. Yang J, Parikh D, Batra D (2016) Joint unsupervised learning of deep representations and image clusters. In: Proceedings of the

IEEE computer society conference on computer vision and pattern recognition, pp 5147–5156
19. Caron M, Bojanowski P, Mairal J et al (2019) Unsupervised pre-training of image features on noncurated data. In: Proceedings of the IEEE international conference on computer vision, pp 2959–2968
20. Huang J, Gong S, Zhu X (2020) Deep semantic clustering by partition confidence maximisation. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp 8846–8855
21. Ji X, Vedaldi A, Henriques J (2019) Invariant information clustering for unsupervised image classification and segmentation. In: Proceedings of the IEEE international conference on computer vision, pp 9864–9873
22. Hadsell R, Chopra S, LeCun Y (2006) Dimensionality reduction by learning an invariant mapping. Proc IEEE Comput Soc Conf Comput Vis Pattern Recogn 2:1735–1742
23. Mnih A, Teh YW (2012) A fast and simple algorithm for training neural probabilistic language models. In: Proceedings of the 29th international conference on machine learning, vol 2, pp 1751–1758
24. Oord A van den, Li Y, Vinyals O (2018) Representation learning with contrastive predictive coding. arXiv:1807.03748
25. He K, Fan H, Wu Y, et al (2020) Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp 9726–9735
26. Henaff OJ, Srinivas A, De Fauw J et al (2020) Data-efficient image recognition with contrastive predictive coding. In: 37th international conference on machine learning, pp 4130–4140
27. Chen T, Kornblith S, Norouzi M et al (2020) A simple framework for contrastive learning of visual representations. arXiv:2002.05709v3
28. Van Gansbeke W, Vandenhende S, Georgoulis S et al (2020) SCAN: learning to classify images without labels. Lect Notes Comput Sci 20:268–285