IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 2 June 2023, accepted 15 June 2023, date of publication 19 June 2023, date of current version 23 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3287771

RESEARCH ARTICLE

Class-Incremental Learning Based on Big Dataset Pre-Trained Models

BIN WEN AND QIUYU ZHU, (Member, IEEE)
School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China

Corresponding author: Qiuyu Zhu (zhuqiuyu@staff.shu.edu.cn)

This work was supported by the Shanghai Science and Technology Innovation Fund under Grant 22511103200.

ABSTRACT Deep neural networks have shown excellent performance in the field of pattern classification and are widely used. However, real-world data are often cannot be obtained at once, and the knowledge of old classes will be heavily forgotten when training new classes of data on the network, which is called catastrophic forgetting. Therefore, the incremental learning method to solve this problem came into being. In this paper, we propose a class-incremental learning method based on a big data pre-trained model, which makes full use of the large amount of public knowledge in the pre-trained model's front network to reduce the forgetting problem of the network in subsequent classification tasks. On the basis of our previous incremental learning method based on PEDCC, we discuss the effects of different pre-trained models, training strategy, training hyperparameters, etc. PEDCC-Loss is used to constrain the cosine distance between the latent feature and the pre-defined class center, and finally the joint prediction is determined by multiple network prediction results. The algorithm in this paper is verified on the CIFAR100, Tiny ImageNet, and FaceScrub datasets with and without partial retention of old samples, and achieves the best results compared to the previous typical class-incremental learning methods. The performance in coarse-grained datasets even exceeds the accuracy of non-incremental learning without pre-trained model. Code is available in https://github.com/byBinWen/Class-Incremental-Learning-Based-on-Big-Dataset-Pre-trained-Models.

INDEX TERMS Incremental learning, image classification, ensemble learning, PEDCC-Loss.

I. INTRODUCTION

The human visual perception system is incremental in nature, and it can keep learning new knowledge while retaining the previously learned knowledge. For example, when learning the letter DEF, human will not forget the previously learned letter ABC. Most of the current pattern recognition systems use all the data for training at once, and acquire the ability of classification according to the label of the training data. However, in practical applications, all the training data may not be obtained at once. Therefore, a more flexible strategy is needed to dynamically process the data obtained in batches in real life for training, so that the network learns new data without forgetting what it has learned before.

However, when the pattern classifier trains data in batches, the performance of the previous task will degrade. This phenomenon is called catastrophic forgetting. Therefore, the main challenge of current incremental learning is how to reduce or avoid catastrophic forgetting, so that the network can achieve performance close to that of training all the data at once when training different batches of data.

Traditional incremental learning methods, such as ensemble learning [1], train multiple single learning models, and then combine them to obtain a unified integrated learning model, to achieve more accurate, more stable and stronger results. Nowadays incremental learning more often uses convolutional neural networks, such as LwF (learning without forget) [2], which proposes an incremental training method based on convolutional neural networks where the convolutional layer parameters are shared at each step and only the linear layer is different. When a new class arrives, the linear layer is expanded by using the distillation loss function and fine-tuning method to save the previously learned knowledge. Therefore, when learning new classes,

The associate editor coordinating the review of this manuscript and approving it for publication was Felix Albu.

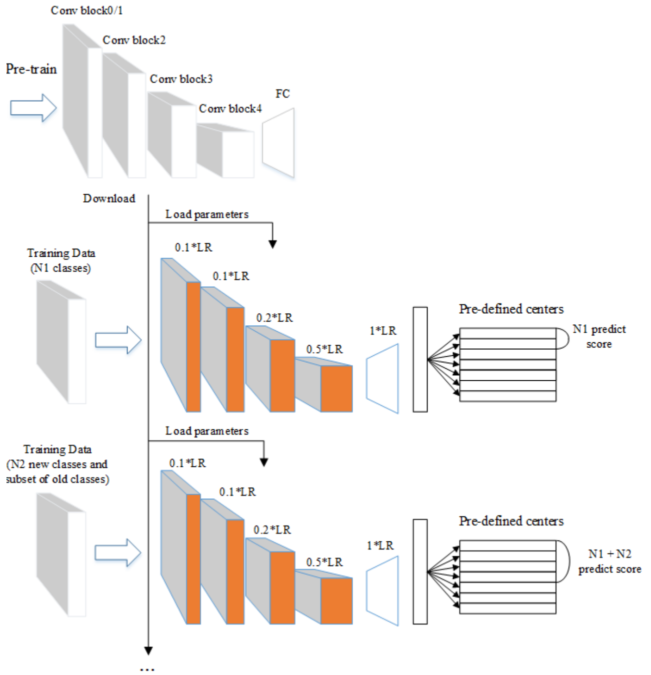

FIGURE 1. Training phase. The new batch of data and the retained part of the old samples (optional) are combined as the training datasets for the training of the new network. The weights from the pre-trained model are used as the initial values, and different learning rates are used for different layers to retain the classification accuracy of the old classes as much as possible. In the figure, LR refers to the overall learning rate of the network in the training process.

the loss function is important to retain the previous class knowledge. The class-incremental learning method based on deep learning, such as iCaRL [3], usually adds the new class samples and the previously stored old class samples to the convolution neural network for training, to update the current model parameters. Rebalance [4] combines cosine normalization, less forgetting constraints and inter-class separation to reduce the adverse impact of the imbalance between the new and old classifiers, and effectively rebalance the training process.

This paper proposes a new incremental learning method based on the big dataset pre-trained model and cosine distance. Because the pre-trained network trained by big dataset contains a large amount of learned knowledge, if we use a lower learning rate for the previous convolutional layer and a higher learning rate for the subsequent convolutional and linear layers to fine-tune the network, we can greatly utilize the common knowledge in the pre training model, thereby making the model suitable for the tasks that need to be applied. At the same time, on the basis of our previous incremental learning architecture based on PEDCC (Predefined Evenly Distributed Class Centroids) [5], we use PEDCC to fix the weights of the last linear layer of each network, so that the output features of different classes are mapped to the predefined class centers respectively, allowing the new class and the old class not to interfere with each other. The final accuracy of incremental learning is significantly improved. In the testing phase, the norm value and the cosine confidence of the network output features are used to determine the selection of the networks. The structure of the training phase is shown in Fig.1.

The contributions of this article are summarized as follows:

1) A class-incremental learning method based on the big dataset pre-trained model is proposed, and the selection criteria of the pre-trained model are given according to the characteristics of different data to be classified.
2) The training strategy of the incremental learning integrated network is designed, and the incremental learning performance with and without old sample retention is verified by retaining old samples with different percentages, which shows that the method in this paper has achieved better performance than the previous methods in both cases.
3) The optimal learning rate weights of different levels of the pre-trained model are obtained through experiments.

II. RELATED WORK

A. CLASS-INCREMENTAL LEARNING BASED ON DEEP LEARNING

At present, the mainstream class-incremental learning methods are mainly divided into three categories. The first is to use growable networks. For example, Xiao et al. [6] propose a network that can grow hierarchically. Each node is composed of clusters of similar classes. Through the tree structure, only some parts of the model need to be adjusted when the model is updated, and the adjustment range of the model can be strictly controlled. Incremental learning is achieved through the growth of the network, but it faces the increasing difficulty of training large networks and the difficulty of how to improve the network capacity effectively. DEN [7] is trained in an online manner by performing selective retraining, dynamically expands network capacity with only the necessary number of units which achieves good performance with substantially fewer number of parameters.

The second category is based on generated samples, such as the phantom samples generated by GAN introduced by V enkatesan [8], to retain the information of the original training samples. These phantom samples and incremental samples are used to train the new deep network, and better class-incremental training results are achieved. However, this method requires additional steps to generate samples, making the process more complex, and it is difficult to apply to the new incremental sample of the old classes.

The third category is from the perspective of improving the loss function. For example, Li [2] propose a method called LwF, which uses the distillation loss function, classification loss function and fine tuning to retain the original model knowledge in the training new classes. The distillation loss function is used to make the output of the new classes close to the output of the original network trained by the previous classes, to maintain the information learned by the original network. By setting the ratio of classification

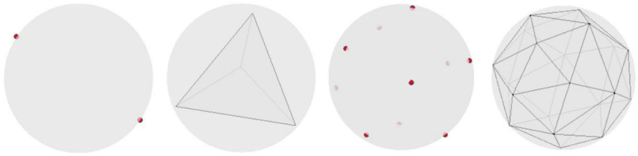

FIGURE 2. The diagram of PEDCC predefined centers. The figure shows 2, 4, 10 and 20 predefined centers generated by PEDCC in 3D space.

loss to distillation loss, it is possible to control whether the data is more inclined to the old classes or the new classes. Hou et al. [4] use the improved distillation loss function and less forgetting loss function to maximize the distance between the new and old classes, to reduce the catastrophic forgetting when training new classes. Minsoo [9] minimizes the upper bound of the loss increases incurred by model updates using the representations, which exploits the estimated importance of each feature map within the model.

In addition, there are also some studies that combine multiple categories of methods, such as Rebuffi et al. [3] use convolutional neural networks for feature learning and representation. The new classes samples and the previously stored old classes samples are added to the convolution neural network for training, to update the current model parameters and obtain the new feature representation of all classes. In the classification step, NCM is used to classify the feature vectors extracted from the sample set. Wu et al. [10] redefined the loss function (cross entropy loss function + distillation loss function) based on iCaRL, and added GANS [11] to generate some samples of old classes, further improving the generalization ability.

B. CLASS-INCREMENTAL LEARNING BASED ON PEDCC-LOSS

PEDCC-Loss [12] (Pre-defined Evenly Distributed Class Centroids Loss) is a classification loss function based on PEDCC, which can make features of different classes have the maximum inter-class distance and the minimum intra-class distance, thus achieving good classification performance. PEDCC-Loss has predefined centers for the features of each class, and the predefined centers are distributed on the feature hypersphere. The distribution diagram of predefined centers in 3D space is shown in Fig.2. These points can be generated based on the physical model [13] with the lowest charge energy on the hypersphere or mathematical formula [14], and PEDCC is also used in POD Loss [15].

The author [5] had proposed an integrated incremental network method based on the PEDCC-Loss, whose multi-network architecture is same as Fig. 1 except for the pre-trained model, learning rates and learning strategy etc. In each batch training, a new sub-classification network that inherits from the previous classification network are added, and trained by the data of the new classes at different learning rates for different layers. In this way, the old network can retain the old knowledge while the new network can learn the new knowledge. After all are completed, a number of networks will be obtained to retain the new and old knowledge respectively. For the retention of old data, a random selection of old samples is used.

For the testing part, PEDCC-Loss is used to constrain the cosine distance between the output features and their corresponding predefined class centers, and the probability representation of the network prediction is converted into the confidence representation based on the cosine distance, so that the nearest result is regarded as the classification result of the network. In the multi-network test, when the class of the test sample is in the training batch of the current network, the norm value of its output features will be larger than that of other networks, so the product of the norm value of the sample features before normalization and the cosine distance is used as the final prediction confidence. The confidence is calculated as follows:

$$C_n = \max_i g_{ni}(x) \cdot \|\mathbf{z}\| \tag{1}$$

where $g_{ni}(x)$ is the i-th cosine distance in the network n, $\|\mathbf{z}\|$ is the latent feature before normalization, C_n is confidence score when sample x is recognized by the network n. The final selected network J is the network with the highest confidence score:

$$J = \operatorname*{argmax}_n C_n \tag{2}$$

The recognized label with the maximum cosine distance between the latent feature and the predefined centers in network J is the final prediction result.

Compared with the traditional method using SoftMax Loss, using PEDCC-Loss and cosine distance metrics could significantly improve the performance of multi-network classification.

C. PRE-TRAINED NETWORK MODEL AND ITS APPLICATION IN INCREMENTAL LEARNING

The selection of pre-trained models is particularly important when using pre-trained models for incremental training. Pre-trained models usually include supervised pre-trained models and self-supervised pre-trained models. Supervised pre-trained models are obtained from the training of labelled datasets, and the obtained features are more dependent on the identification characteristics of the datasets; The self-supervised pre-trained models are trained by setting a self-supervised learning strategy on the data without class labels, which can obtain more extensive image features.

For the problem of self-supervised learning, Chen et al. [16] propose SimCLR, introduce learnable nonlinear transformation between the representation and contrastive loss to improve the quality of learned representation, and use larger batch size and more training steps in the training part to achieve good self-supervised learning classification accuracy. Zbontar et al. [17] take a different perspective by proposing a Barlow Twins approach that measures the cross-correlation matrix between the outputs of two identical networks that use distorted versions of samples and make it as close to the identity matrix as possible.

For incremental learning methods based on pre-trained models, Yang et al. [18] propose a continuous learning Bayesian generative model built on a fixed pre-trained feature extractor, where the knowledge of each old class can be compactly represented as an ensemble of statistical distributions, using a Gaussian mixture model to avoid forgetting in continuous learning. However, the pre-trained feature extractor of this method is trained on the first batch of the classification datasets, whose features are not extensive and effective enough, and will remain unchanged later, therefore limiting the performance of incremental learning.

Huang et al. [19] theoretically study the reason why self-supervised learning has excellent generalization ability on downstream tasks. It shows that the generalization ability is related to three key factors: alignment of positive samples, divergence of class centers, and concentration of augmented data. SimCLR and Barlow Twins just fit the factors.

In this paper, the pre-trained model learned from ImageNet big datasets is used to enhance the effectiveness of features, and then the model will be continuously optimized with incremental learning, resulting in better incremental learning performance.

III. CLASS-INCREMENTAL LEARNING BASED ON PRE-TRAINED MODELS AND PEDCC

A. SELECTION OF PRE-TRAINED MODELS

Since different levels of convolutional kernels in a convolution neural network learn different information, the higher the level, the stronger the semantic information [20], [21]. According to [21] the features output from the bottom layers of the convolution neural network are some highly reusable line and color information, while the features output from the next layers are the contour, shape and other information combined by these bottom features. Due to the high reusability of the bottom layers, we believe that compared with high-level semantic information, retaining more bottom layer information is conducive to incremental learning to a certain extent.

An ideal feature extractor should output two different feature vectors when two input data are visually different, and the more visually similar the input, the more similar the feature vectors obtained from the feature extractor will be. The visual feature extractor (i.e., visual pathway) of young infants may be taught through some way of self-supervision, although the self-supervision mechanisms of infant brain have not been clearly understood [22].

Pre-trained models are usually obtained from large datasets such as ImageNet through extensive training. They have rich visual information and can provide good feature extraction in most image recognition tasks. Therefore, using this feature, this paper uses a lower learning rate for the first several layers of the pre-trained network, that is, retains most of the existing feature extraction functions, reduces the change of common knowledge, and trains the later layers and the linear layer with a higher learning rate, so that they can better fit the current image classification task. The setting of learning rates for different network layers will be discussed in later experiments.

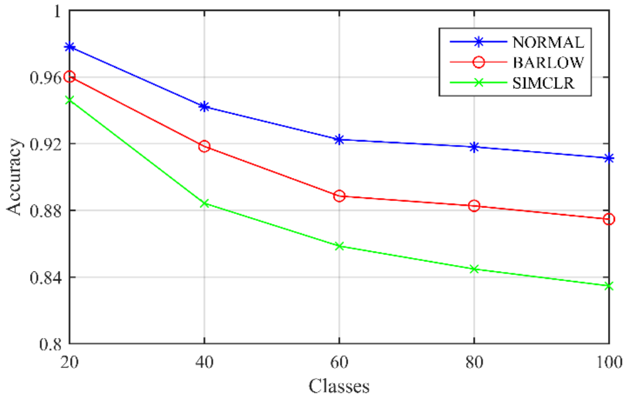

FIGURE 3. Comparative experimental results on FaceScrub. Each step has 20 classes. The supervised pre-trained model that comes with the Pytorch framework achieves the best performance.

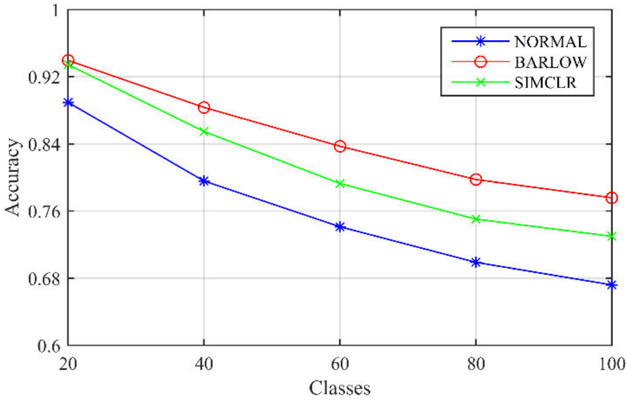

FIGURE 4. Comparative experimental results on CIFAR100. Each step has 20 classes. Barlow's self-supervised pre-trained model achieves the best performance.

In order to compare the impact of different pre-trained models on the classification accuracy of incremental learning, the three pre-trained models selected in this paper are all of the Resnet-50 structure. NORMAL represents the supervised pre-trained model that comes with the Pytorch framework used in this experiment, and BARLOW represents the self-supervised pre-trained model provided by Barlow Twins studied by Zbontar et al. [16], and SIMCLR represents the self-supervised pre-trained model provided by SimCLR studied by Chen et al. [15]. Here, the three datasets are trained in five incremental learning steps (each step has 20 classes for FaceScrub [23] and CIFAR100 [24] datasets, and has 40 classes for Tiny ImageNet [25] datasets), and comparative experimental results are shown in Figs.3, 4, and 5.

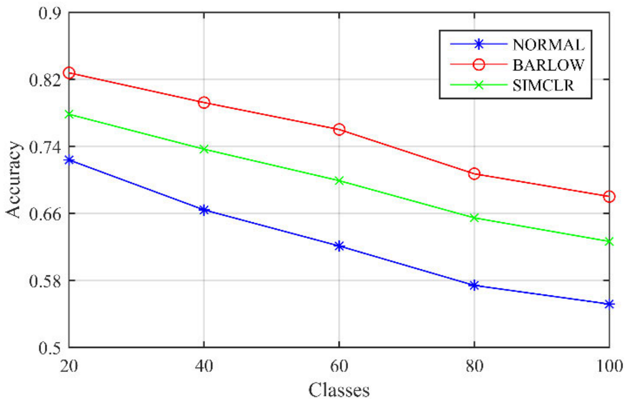

FIGURE 5. Comparative experimental results on Tiny ImageNet. Each step has 40 classes. Barlow's self-supervised pre-trained model achieves the best performance.

It can be seen from Figs.3, 4 and 5 that the optimal pre-trained model is different for different datasets. For FaceScrub datasets, the best classification accuracy is achieved using the supervised pre-trained model that comes with the Pytorch framework, while for the CIFAR100 and Tiny ImageNet datasets, Barlow's self-supervised pre-trained model can achieve the best classification accuracy. This may be due to the fact that for fine-grained classification tasks such as face datasets, supervised training using datasets with labels can enable the network to learn the subtle difference information that is distinguished by labels in similar samples; For coarse-grained datasets such as CIFAR100 and Tiny ImageNet, especially when the pre-trained network is trained with ImageNet datasets, the self-supervised training method can extract the information that would not be noticed when manually labeling, which is conducive to further improving the classification accuracy.

When selecting the experimental datasets, the supervised pre-trained model should not be used for the samples of the trained label classes; For the self-supervised pre-trained model, there is no restriction on the selection of experimental datasets. In the following, unless otherwise specified, the supervised pre-trained model is used for FaceScrub, and the Barlow self-supervised pre-trained model is used for CIFAR100 and Tiny ImageNet.

B. TRAINING STRATEGY

In the training phase of this paper, the system flow in Fig.1 is used. For a new batch of class data, train a new network based on PEDCC-Loss, so that the knowledge of different batches of training data is retained in different networks, and the final classification result is based on the highest confidence score in Eq. 2.

First, according to the application scenario, the maximum number of classes N and the number of dimensions M of the predefined class centers are set, and the corresponding PEDCC points are generated [13], that is, a PEDCC matrix with N rows and M columns. The weight of the last classification layer of the neural network is initialized by the PEDCC centers and will not change during the training process, as shown in Fig.1. For example, when N1 = 20, M = 512, the training set of the first neural network is the first 20 class samples, and the first 20 predefined class centers are used. After the training is completed, the network model is retained.

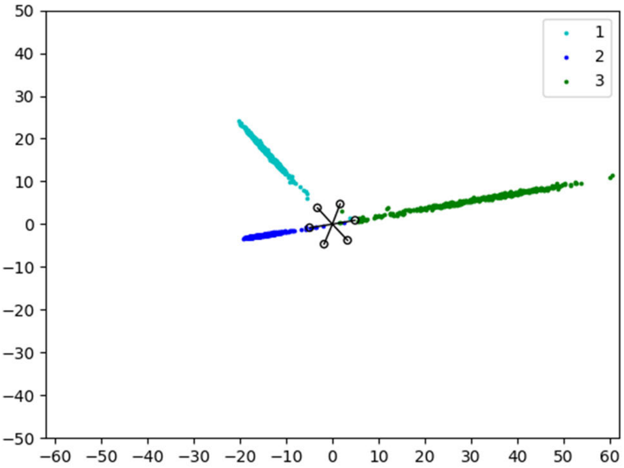

FIGURE 6. Features are constrained to three of six predefined evenly distributed class centroids in a two-dimensional space in the first batch of training.

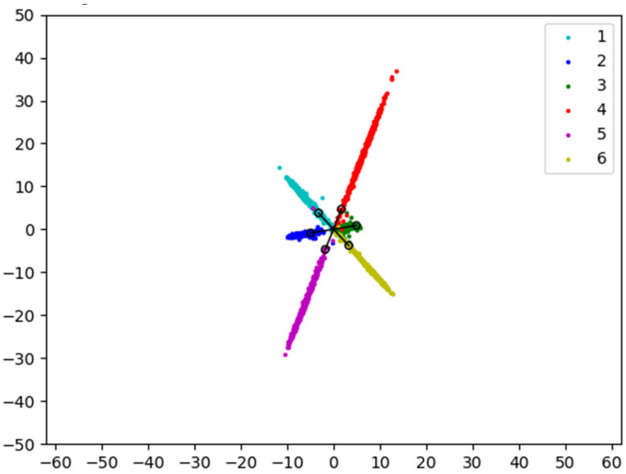

FIGURE 7. Features are constrained to six predefined evenly distributed class centroids in a two-dimensional space in the second batch of training.

As shown in Fig.6 Fig.7, a two-step incremental learning process for a 6-class dataset (part of EMNIST) is demonstrated. In order to display the image, the dimension of the feature space is set to 2, which means that 6 evenly distributed centroids (denoted as black circle) are predefined in a 2-dimensional space. Three of them are used in the first batch of training, and all six of them are used in the second training.

From the figures, it can be seen that the centroids of the classes learned in the first batch of training have not changed after the second batch of training, which provides a basis for multi-network classification.

Noted that according to the PEDCC theory, when $N \geq M + 1$, N points could be evenly distributed in a M-dimensional space. So up to three points could be evenly distributed in a two-dimensional space. Here six points are only used for demonstration, although it seems like that the distance between some two points is not truly equal in the figures. In actual experiments, the feature space has 512 dimensions, which can ensure that the distance between any two points is equal when the datasets just have 100 or 200 classes.

Since the pre-trained models are introduced in this experiment for the initial knowledge of incremental learning, the knowledge of the neural network for the specific classification task requirements largely comes from the existing large amount of knowledge in the pre-trained model, so for the selection of the initial network for each batch of training, two training strategies are compared. First, each batch of training starts from the pre-trained model; Second, the first batch of training starts from the pre-trained model, and the subsequent batch of training starts from the network completed in the previous batch, that is, the strategy in [5]. Here, three datasets are tested separately and trained in five steps (each step has 20 classes for FaceScrub and CIFAR100 datasets, and has 40 classes for Tiny ImageNet datasets). The results are shown in Table 1. In this experiment, for the subsequent batches of the training sets, a retention strategy of 8% random retention is adopted for the old samples.

TABLE 1. Comparison of two training strategies for three datasets with 8% of old samples retained at random.

Datasets and Strategy	one batch	two batches	three batches	four batches	five batches
FaceScrub-ind	97.82%	94.22%	92.25%	91.81%	**91.14%**
FaceScrub-con	98.35%	91.20%	79.13%	75.48%	70.79%
CIFAR100-ind	93.95%	88.35%	83.72%	79.76%	**77.58%**
CIFAR100-con	93.50%	87.03%	81.53%	76.85%	73.22%
Tiny ImageNet-ind	82.80%	79.25%	76.03%	70.74%	**68.03%**
Tiny ImageNet-con	82.50%	75.90%	68.92%	61.96%	57.65%

In Table1, "Ind" denotes the first strategy, where each training starts from the pre-trained model and the networks are independent; "Con" denotes the second strategy, where the first training starts from the pre-trained model and each subsequent training starts from the network where the previous training was just completed.

From the experimental results in Table 1, it can be found that after the first batch of training, regardless of the dataset, there is basically no difference in the classification accuracy of the two strategies for the first batch of samples, because the differences between the two strategies are not yet reflected at this time. Starting from the second batch, the second training strategy will lead to a faster decrease in training accuracy, especially on CIFAR100 and Tiny ImageNet datasets. This is because the main knowledge about classification tasks comes from pre-trained models. If each training continues from the last trained model, due to some changes in the weights of the former level of the network, it may lead to a decrease in performance in subsequent classification tasks. It is better to start training directly from the pre-trained model.

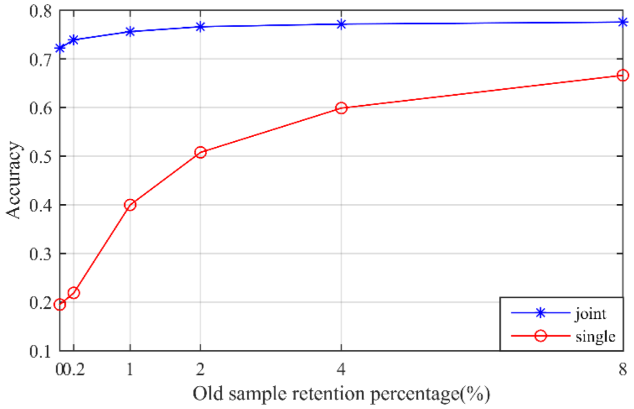

FIGURE 8. The impact of retention percentage of old samples on classification performance.

For this reason, it may also be possible to obtain good classification performance without retaining any old samples in subsequent batches. To verify this case, a test is conducted on the CIFAR100 for 5 steps, as shown in Fig.8.

In Fig.8, "joint" denotes the joint prediction, and "single" denotes the last network prediction. It can be found that as the retention percentage of old samples decreases, the classification performance of one single network sharply decreases. This is because the last network cannot obtain knowledge of the previous classes, but the classification performance of multi-network joint prediction only slightly decreases because the samples of the previous classes are mainly classified by the previous networks. Therefore, this can fully prove that when using multiple network structures and pre-trained models, the dependence on retained old samples is very low.

In addition, due to the fact that only 8% of the old samples are retained, and considering the imbalance in the proportion of new and old samples, an attempt is made to adjust the weight of the loss of the old samples, as shown in Fig.9. From the figure, it can be seen that due to multiple network predictions, focusing too much on other batch data can actually lead to performance degradation. Therefore, the weight of loss of old samples is not controlled in this experiment.

Similarly, the same tests are conducted on the three datasets without retaining old samples to verify the first strategy is better, which is shown in Table 2.

Comparing Table 2 with Table 1, the classification accuracy is somewhat degraded in both cases, but still has good performances. And the first strategy is still better.

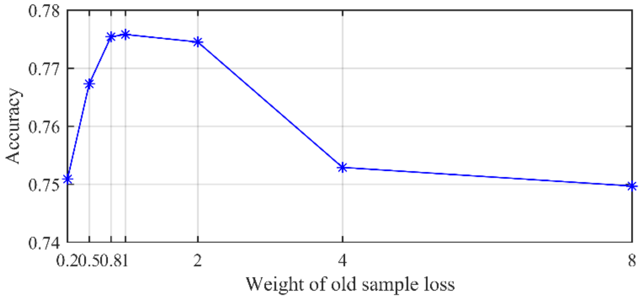

FIGURE 9. The impact of the weight of old sample loss on classification performance.

TABLE 2. Comparison of the two training strategies for three datasets without retaining the old samples.

Datasets and Strategy	one batch	two batches	three batches	four batches	five batches
FaceScrub-ind	98.65%	95.82%	93.40%	92.29%	**91.91%**
FaceScrub-con	98.85%	92.48%	86.12%	81.78%	78.29%
CIFAR100-ind	94.20%	84.88%	79.38%	75.69%	**72.21%**
CIFAR100-con	93.65%	83.68%	77.00%	72.73%	68.12%
Tiny ImageNet-ind	82.50%	77.23%	73.45%	68.68%	**65.44%**
Tiny ImageNet-con	83.00%	74.48%	67.35%	59.94%	55.67%

Therefore, subsequent experiments will be conducted from two ways, incremental learning following an 8% random old sample retention strategy, and incremental learning with no old samples retained at all where only new samples trained per batch.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. EXPERIMENTAL SETUP AND DATASETS

The experimental datasets used in this paper are CIFAR100 [24] (100 classes), Tiny ImageNet [25] (200 classes), and FaceScrub [23] (100 classes). CIFAR100 has 100 classes with an image resolution of 32 × 32. The number of training and testing images for each class is 500 and 100 respectively. Tiny ImageNet has 200 classes with an image resolution of 64 × 64 and 500 training and 50 testing samples per class. FaceScrub has 100 classes with an image resolution of 64 × 64 and approximately 160 training and 40 testing samples per class. The network model used in this experiment is obtained by modifying the last fully connected layer based on ResNet50 and with the addition of a fixed weight PEDCC classification layer, as shown in Fig.10.

The size of the convolutional kernel is consistent across the convolutional layers at 3 × 3, with stride and padding of 1. We use the Pytorch1.4 framework to train our neural network with 100 epochs. The learning rate starts at 0.1 and is divided by 10 after 30, 60, and 90 epochs.

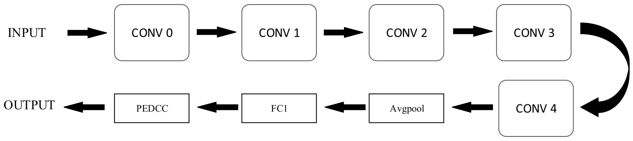

FIGURE 10. Network structure based on ResNet50.

SGD are used to train the network with a weight decay parameter of 0.0005 and a momentum of 0.9. This experiment was compared with experiments by Zhu [5], AFC [9], Yang [18], UCIR [4], and iCaRL [3], and with the results of training the full dataset at once with and without a pre-trained model.

B. IMPACT OF LEARNING RATE WEIGHTS

Due to the use of the pre-trained network as the feature extractor, it is not advisable to change the parameters of the first few levels of the network too much. To test the impact of different convolutional layer learning rate weights on incremental learning classification accuracy, experiments are conducted, as shown in Table 3. A supervised pre-trained model is used for FaceScrub datasets, with a batch size of 64 and trained in 5 steps (i.e., class-incremental step size of 20 classes).

TABLE 3. Comparison of learning rate weights for different layers in FaceScrub 5-step training.

Conv0/1	Conv2	Conv3	Conv4	ACC-re	ACC-no-re
0.1	0.1	0.2	0.5	**91.14%**	**91.91%**
0.1	0.2	0.2	0.5	89.68%	91.58%
0.2	0.2	0.5	0.7	87.53%	89.37%
0.1	0.1	0.2	0.2	90.53%	91.81%
0.2	0.2	0.5	0.5	88.81%	89.73%
0.1	0.2	0.5	0.5	89.22%	89.14%

ACC-re here represents the classification accuracy when retaining old samples, and ACC-no-re represents the classification accuracy when not retaining old samples. In the experimental results in Table 3, when the learning rate weight of the linear layer is 1, and the learning rate weights of Conv0/1, Conv2, Conv3, and Conv4 are set to 0.1, 0.1, 0.2, and 0.5, respectively, the classification accuracy is optimal, and this conclusion holds true for both the case of retaining old samples and the case of not retaining old samples. In the following experiments, learning rate weights of 0.1, 0.1, 0.2, and 0.5 are used in all cases.

C. EXPERIMENTAL RESAULTS AND DISCUSSION

In the experiment, incremental learning is compared with the training results of all datasets of non-incremental learning, and the baseline is divided into two types, one is using a pre-trained model, labeled pre-base, and one training from scratch without using a pre-trained model, labeled no-pre-base. The full datasets trained with the pre-trained model is trained as normal training with the same learning rate for all layers.

1) EXPERIMENTAL RESULTS AND DISCUSSION ON CIFAR100 DATASET

Table 4 shows the experimental results of this method for 25 and 20 classes of each step for CIFAR100, as well as a comparison with the entire dataset training. Fig.10 shows a comparative test of this method with Zhu, AFC, Yang, UCIR, and iCaRL.

As can be seen from Table 4 and Fig.11, using the self-supervised pre-trained model for CIFAR100 can significantly improve the classification accuracy of the entire dataset, from 74.86% to 83.21%. Furthermore, the classification accuracy of incremental learning using the self-supervised pre-trained model in this paper is significantly improved compared to incremental learning methods without pre-trained models (Zhu, AFC, UCIR, etc.), and even exceeds the non-incremental learning results without pre-trained models. Compared with the method proposed by Yang, which uses pre-trained models, this method using multi-network structure, achieves better performance. In addition, there is a difference of about 5% in classification accuracy between retaining old samples and not retaining old samples, but the result of not retaining old samples also exceeds the classification accuracy of Zhu's method of retaining old samples thanks to the introduction of pre-trained models.

TABLE 4. Experimental results on CIFAR100.

Number of classes per batch	one batch	two batches	three batches	four batches	five batches
25-re	93.28%	86.52%	81.32%	79.67%	
25-no-re	93.00%	83.72%	77.05%	74.10%	
20-re	93.95%	88.35%	83.72%	79.76%	**77.58**%
20-no-re	94.20%	84.88%	79.38%	75.68%	72.21%
Pre-base					83.21%
No-pre-base					74.86%

Due to the fact that the initial knowledge of the network comes from common knowledge in pre-trained models, the order of the data during training has little impact on performance. Ordered and several unordered CIFAR100 datasets are tested, and the results show that the order of the data may have some impact on classification accuracy at the first few batch data, because some classes which are difficult or easy to be classified may be divided into the same batch. But for the entire data, the standard deviations σ of classification accuracy are 0.41 for retained old samples and 0.4 for no retained old samples, which means that thanks to the initial knowledge provided by pre-trained models, the order of data has little impact on accuracy of the entire data.

Table 5 shows the classification performance of different batch data individually after the entire training process.

TABLE 5. Classification accuracy of different batch data on CIFAR100 datasets.

Classes	1-25	26-50	51-75	76-100	
25-re	86.24%	79.48%	73.72%	79.24%	
25-no-re	73.80%	74.12%	69.12%	79.36%	
Classes	1-20	21-40	41-60	61-80	81-100
20-re	85.40%	83.20%	76.70%	69.70%	72.90%
20-no-re	72.30%	73.85%	74.20%	67.95%	72.75%

From Table 5, it can be seen that the joint classification ability of multi-networks greatly reduces catastrophic forgetting. For cases where old samples are not retained, the standard deviations of the accuracy of different batch data are 3.6 for 4 steps and 2.2 for 5 steps, which are smaller than 4.4 for 4 steps and 6.0 for 5 steps when retaining old samples, and the difference between accuracies of different batch data is caused by the classification difficulty of the current batch data. In other experiments on unordered datasets, the results changed, but the performances are always stabler. For cases where some old samples are retained, the classification accuracy of old classes is higher. This is because the network could keep more data on old classes when learning samples of old classes, achieving better performance on old classes. As the number of training classes increases, there is a trend of accuracy decrease, but it is also affected by the classification difficulty of the batch data. In addition, the confusion matrix for different batch data in 5 incremental steps of CIFAR100 datasets is given, as Table 6.

TABLE 6. Confusion matrix for different batch data in 5 incremental steps of CIFAR100 datasets.

Classes	1-20	21-40	41-60	61-80	81-100
1-20	1769	99	68	33	31
21-40	173	1702	60	30	35
41-60	154	156	1587	34	69
61-80	205	194	120	1435	46
81-100	188	160	114	59	1479

2) EXPERIMENTAL RESULTS AND DISCUSSION ON TINY IMAGENET DATASET

Table 7 shows the experimental results of this method for each step of 50 and 40 classes for Tiny ImageNet, as well as a comparison with the entire dataset training. Fig.11 shows a comparative test of this method with Zhu, AFC, UCIR, and iCaRL.

As can be seen from Table 7 and Fig.12, the experimental results for Tiny ImageNet are similar to those of CIFAR100. The application of the pre-trained model can improve the classification accuracy of the integer dataset training from 59.04% to 73.63%, and the classification accuracy of incremental training can also be greatly improved. This is because the pre-trained model contains a large amount of

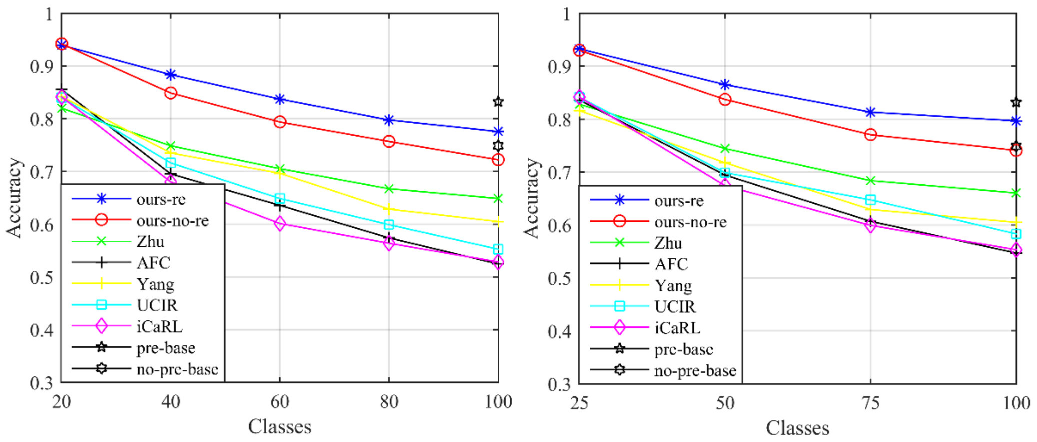

FIGURE 11. Experimental results for CIFAR100. On the left, class-increment step size is 20, and on the right, class-increment step size is 25.

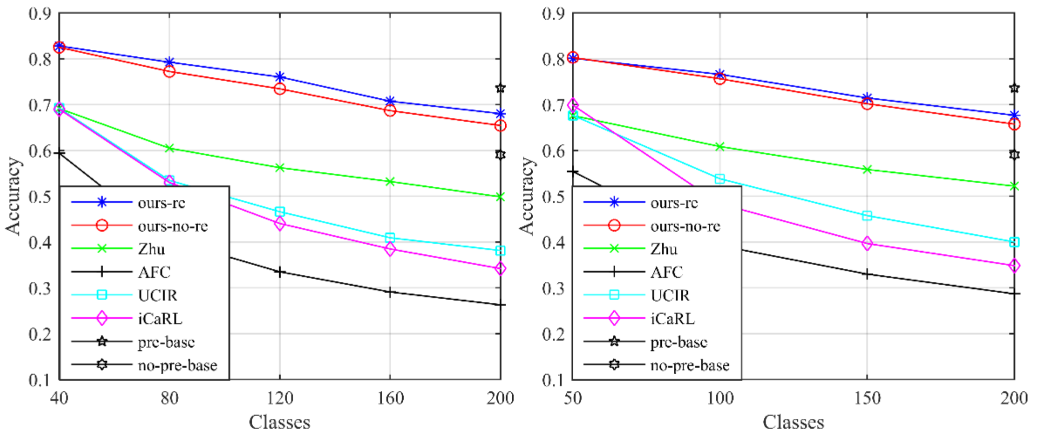

FIGURE 12. Experimental results on Tiny ImageNet. On the left, class-increment step size is 40, and on the right, class-increment step size is 50.

TABLE 7. Experimental results on tiny ImageNet.

Number of classes per batch	one batch	two batches	three batches	four batches	five batches
50-re	80.12%	76.60%	71.45%	67.67%	
50-no-re	80.28%	75.64%	70.19%	65.74%	
40-re	82.80%	79.25%	76.03%	70.74%	**68.03%**
40-no-re	82.50%	77.23%	73.45%	68.68%	**65.44%**
Pre-base					73.63%
No-pre-base					59.04%

common knowledge, which has a good effect on feature extraction for subsequent classification tasks.

For incremental learning, the front level network contains a large amount of knowledge, which helps reduce changes in low-level visual feature extraction capabilities when training data changes, and is conducive to improving overall classification accuracy. The difference between retaining old samples and not retaining old samples is about 2%, which is smaller than the difference in CIFAR100. This may be because the Tiny ImageNet is larger, and there are more sample classes for the class increment step size (for example, there are 40 classes in 5stepscompared to 20 classes in CIFAR100). The richer training samples enable each batch of training to learn more rich knowledge. At this time, whether to retain the old samples brings about the "review" effect of the old knowledge is not so important.

There is another indicator in class-incremental learning. Forgetting rate F (proposed by Liu [26]) is calculated by the accuracy difference between the first and the last network for the first batch of tasks, as follows:

$$F = A_N^Z - A_0^Z \tag{3}$$

where A_i^Z is the average accuracy of the first batch data predicted by the i-th network. This indicator shows the forgetting of the first batch data by the network after N steps training, so the lower forgetting rate is better.

Two datasets above are compared with other methods, as shown in Table 8.

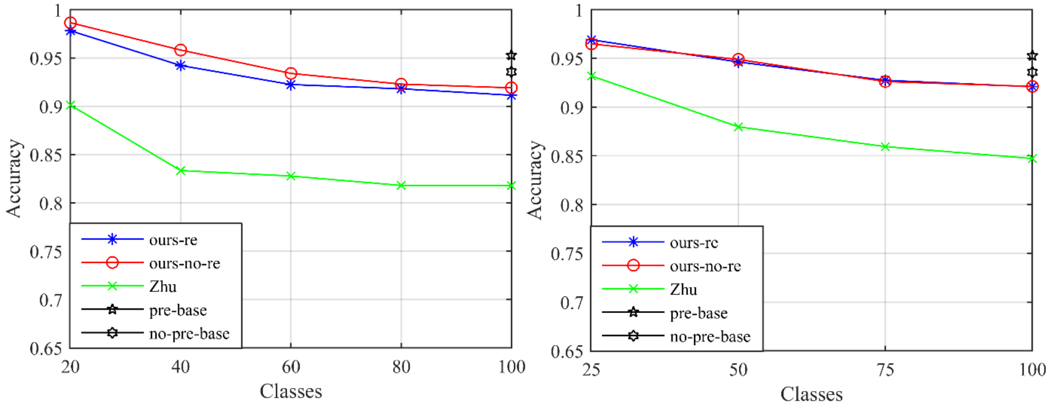

FIGURE 13. Experimental results on FaceScrub. On the left, class-increment step size is 20, and on the right, class-increment step size is 25.

TABLE 8. Experimental results of forgetting rate in five steps.

Method	CIFAR100	Tiny ImageNet
Ours-re	**8.55%**	**7.3%**
Ours-no-re	18.52%	16.25%
Zhu	12.34%	10.31%
AFC	29.35%	12.89%
Yang	24.21%	/
UCIR	18.70%	31.88%
iCaRL	31.88%	43.40%

TABLE 9. Experimental results on FaceScrub.

Number of classes per batch	one batch	two batches	three batches	four batches	five batches
25-re	96.91%	94.63%	92.75%	92.09%	
25-no-re	96.49%	94.89%	92.62%	92.12%	
20-re	97.82%	94.22%	92.25%	91.81%	91.14%
20-no-re	98.65%	95.82%	93.40%	92.29%	91.91%
Pre-base					95.28%
No-pre-base					93.56%

In Table 8, the proposed method with retained old samples achieves the best performance. The introduction of pre-trained models allows the network to have more initial knowledge at the beginning of learning. And the multi-network prediction effectively preserves the classification performance on the first task. Considering that 'Ours-no-re' is worse than Zhu, the use of old data is also important for preserving old knowledge. So, the combination of using old data and pre-trained models achieves the best performance.

3) EXPERIMENTAL RESULTS AND DISCUSSION ON FACESCRUB DATASET

Table 9 gives the experimental results of this method for FaceScrub for 25 and 20 classes per step, and the comparison with the integer dataset training. Fig.12 shows the experiments comparing the method in this paper with Zhu.

In the experiments in Table 9 and Fig.13, it can be found that for a fine-grained dataset such as FaceScrub, the use of the pre-trained model has little impact on the training of the integer dataset, with the classification accuracy increasing from 93.56% to 95.28%, but the improvement for incremental learning is still significant. The reason is that small datasets have a small sample size during early training, which makes it difficult for network training to obtain sufficient primary visual knowledge. Therefore, this will also bring greater difficulties in subsequent training when more classes are added. That is, the front network lacks sufficient feature extraction capabilities, resulting in poor performance of subsequent classification tasks. The use of the pre-trained model enables the network to utilize sufficient common knowledge in the pre-trained model's front network during the first training, resulting in better classification accuracy during the first training, and better classification capabilities for subsequent incremental learning.

In addition, regardless of whether the class increment step size is 20 classes or 25 classes, unlike the other two datasets, for FaceScrub, the classification accuracy of not retaining old samples is equivalent to or slightly superior to that of retaining old samples, which indicates that the retention of old samples has almost no contribution to such fine-grained classification tasks, and instead may prevent the current finetuning from better fitting the current batch of classification tasks.

V. CONCLUSION AND DISCUSSION

This paper proposes an integrated class-incremental learning method based on big dataset pre-trained models. In each step of training, the network starts with the pre-trained model with a large amount of common knowledge. We discuss the effects of different pre-trained model, training strategy, training hyperparameters to preserve reusable knowledge, and finally test it with confidence score based on cosine distance and norm values of the features. Experiments on CIFAR100, Tiny ImageNet, and FaceScrub have shown good results. Compared with the methods based on generated samples, proposed method does not require generating samples, making training simpler. Due to the low dependency of this method on old samples, good classification accuracy has also been achieved in experiments that do not retain old samples; Especially, the performance in coarse-grained datasets even exceeds the non-incremental learning accuracy without pre-trained models.

Despite the SOTA results achieved by the method in this paper, class-incremental learning has not yet been fully resolved. In future work, we will explore ways to more efficiently utilize pre-trained models and simplify training processes, and combine OOD detection results of classification networks to achieve better incremental learning performance.

REFERENCES

[1] G. Valentini and F. Masulli, "Ensembles of learning machines," in *Proc. Italian Workshop Neural Nets*. Berlin, Germany: Springer, Sep. 2002, pp. 3–20.

[2] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 12, pp. 2935–2947, Dec. 2018.

[3] S. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "iCaRL: Incremental classifier and representation learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5533–5542.

[4] S. Hou, X. Pan, C. C. Loy, Z. Wang, and D. Lin, "Learning a unified classifier incrementally via rebalancing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 831–839.

[5] Q. Zhu, Z. He, and X. Ye, "Incremental classifier learning based on PEDCC-loss and cosine distance," *Multimedia Tools Appl.*, vol. 80, no. 25, pp. 33827–33841, Oct. 2021.

[6] T. Xiao, J. Zhang, K. Yang, Y. Peng, and Z. Zhang, "Error-driven incremental learning in deep convolutional neural network for large-scale image classification," in *Proc. 22nd ACM Int. Conf. Multimedia*, Nov. 2014, pp. 177–186.

[7] J. Yoon, E. Yang, J. Lee, and S. J. Hwang, "Lifelong learning with dynamically expandable networks," in *Proc. 6th Int. Conf. Learn. Represent.*, Jan. 2018, pp. 1–11.

[8] R. Venkatesan, H. Venkateswara, S. Panchanathan, and B. Li, "A strategy for an uncompromising incremental learner," 2017, *arXiv:1705.00744*.

[9] M. Kang, J. Park, and B. Han, "Class-incremental learning by knowledge distillation with adaptive feature consolidation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 16050–16059.

[10] Y. Wu, Y. Chen, L. Wang, Y. Ye, Z. Liu, Y. Guo, Z. Zhang, and Y. Fu, "Incremental classifier learning with generative adversarial networks," 2018, *arXiv:1802.00853*.

[11] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*.

[12] Q. Zhu, P. Zhang, Z. Wang, and X. Ye, "A new loss function for CNN classifier based on predefined evenly-distributed class centroids," *IEEE Access*, vol. 8, pp. 10888–10895, 2020.

[13] Q. Zhu and R. Zhang, "A classification supervised auto-encoder based on predefined evenly-distributed class centroids," 2019, *arXiv:1902.00220*.

[14] H. Hu, Y. Yan, Q. Zhu, and G. Zheng, "Generation and frame characteristics of predefined evenly-distributed class centroids for pattern classification," *IEEE Access*, vol. 9, pp. 113683–113691, 2021.

[15] Q. Zhu and X. Zu, "A Softmax-free loss function based on predefined optimal-distribution of latent features for deep learning classifier," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 3, pp. 1386–1397, Mar. 2023.

[16] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. 37th Int. Conf. Mach. Learn.*, vol. 119, 2020, pp. 1597–1607.

[17] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, "Barlow Twins: Self-supervised learning via redundancy reduction," in *Proc. 38th Int. Conf. Mach. Learn.*, Jul. 2021, pp. 12310–12320.

[18] Y. Yang, Z. Cui, J. Xu, C. Zhong, R. Wang, and W.-S. Zheng, "Continual learning with Bayesian model based on a fixed pre-trained feature extractor," in *Proc. 24th Int. Conf. Med. Image Comput., Comput.-Assist. Intervent.*, Sep. 2021, pp. 397–406.

[19] W. Huang, M. Yi, X. Zhao, and Z. Jiang, "Towards the generalization of contrastive self-supervised learning," 2021, *arXiv:2111.00743*.

[20] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[21] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 818–833.

[22] F. Ribordy, A. Jabès, P. B. Lavenex, and P. Lavenex, "Development of allocentric spatial memory abilities in children from 18 months to 5 years of age," *Cognit. Psychol.*, vol. 66, no. 1, pp. 1–29, Feb. 2013.

[23] H. Ng and S. Winkler, "A data-driven approach to cleaning large face datasets," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 343–347.

[24] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Tech. Rep., 2009.

[25] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[26] Y. Liu, Y. Su, A.-A. Liu, B. Schiele, and Q. Sun, "Mnemonics training: Multi-class incremental learning without forgetting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12242–12251.

BIN WEN received the bachelor's degree from the School of Electronics and Information Engineering, Tiangong University, in 2020. He is currently pursuing the degree with the School of Communication and Information Engineering, Shanghai University. His research interests include computer vision, pattern recognition, and deep learning.

QIUYU ZHU (Member, IEEE) received the bachelor's degree from Fudan University, in 1985, the master's degree from the Shanghai University of Science and Technology, in 1988, and the Ph.D. degree in information and communication engineering from Shanghai University, in 2006. Currently, he is a Professor with Shanghai University. He is the coauthor of over 100 academic papers and a principal investigator for more than ten governmental funded research projects, more than 30 industrial research projects, many of which have been widely applied. His research interests include image processing, computer vision, machine learning, smart city, and computer application.