Journal of Computing and Electronic Information Management
ISSN: 2413-1660 | Vol. 13, No. 3, 2024

Class Incremental Learning Method Based on Dynamic Structure Extension and Feature enhancement

Zhenghu Li *, Baishun Shi

School of Computer Science and Technology, Qingdao University, Qingdao, Shandong, 266000, China
* **Corresponding author**: Zhenghu Li (Email: 979262157@qq.com)

Abstract: With the advancement and widespread adoption of deep learning models, there has been a growing interest in class incremental learning. This approach aims to continuously learn new classes while retaining the recognition and memory capabilities for previously learned classes within an open and dynamic environment. The primary focus of class incremental learning is on maintaining the ability to learn new classes while mitigating catastrophic forgetting, thus achieving a better balance between stability and adaptability. To address this challenge, we propose an innovative method for incremental class learning that leverages dynamically representations to facilitate more efficient incremental class learning, preserving previously acquired features while adapting to new ones and effectively reducing catastrophic forgetting. Furthermore, we introduce a feature augmentation mechanism to significantly enhance the model's classification performance when incorporating new classes. This approach ensures efficient learning of both old and new classes without compromising the effectiveness of previous models. We conducted extensive experiments on two classes incremental learning benchmarks, consistently demonstrating significant performance advantages over other methods.

Keywords: Machine learning; Incremental learning; Feature enhancement; Structural Expansion.

1. Introduction

Class incremental learning is a key machine learning paradigm that demonstrates enormous application value in responding to dynamic and constantly changing real-world environments. With the rapid development of technology, class incremental learning has found a wide range of applications in multiple fields, especially in cutting-edge intelligent systems such as image classification [1] and natural language processing [2]. The system needs to constantly adapt and learn new data classes to maintain its efficiency and accuracy. This challenge requires the model to simulate the learning process of the human brain, which involves accumulating knowledge in a continuous stream of information and gradually mastering new skills. Therefore, how to design a machine learning mechanism that can learn new classes flexibly like the human brain remains an important issue that urgently needs to be addressed in the scientific research community.

An excellent class incremental learning model should have the ability to absorb new knowledge while firmly retaining the memory of old classes, thus finding a balance between stability and plasticity. Recently, various knowledge replay strategies [3] have emerged in the field of deep networks. Generative replay strategy, as one of the approaches, attempts to create old data samples by training generative models. However, the training process of generative models consumes huge resources, especially when dealing with complex datasets, which require huge memory and computing resources. The parameter regularization method [4] attempts to solve the problem from another perspective by limiting the update amplitude of model parameters to slow down forgetting. Although this method is effective to some extent, excessive constraints may limit the model's ability to absorb new knowledge. With the deepening of research on class incremental learning, more and more people are combining various incremental learning methods and using comprehensive class methods to conduct research.

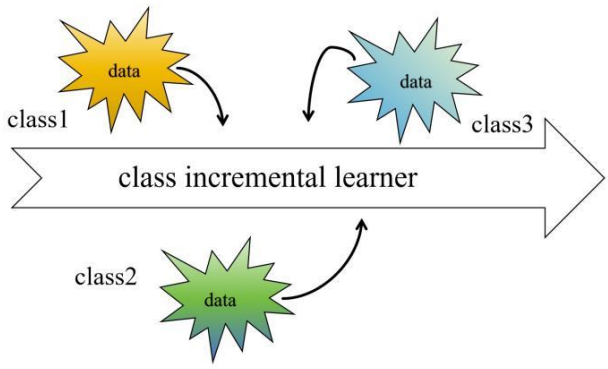

Figure 1. Class incremental learning

In this work, to address the aforementioned weaknesses and better balance the stability and plasticity of class incremental learning, we designed an extensible dynamic framework that utilizes enhanced features of old knowledge to identify features of new classes. We adopt a dynamic architecture that parameterizes and replays features from old classes, while dynamically expanding feature extractors to learn new features from new classes. This new architecture consists of a single primary feature extractor shared across all classes and an advanced feature extractor added at each incremental step. Finally, a unified classifier is input for classification prediction, thereby maintaining the utilization of previous old knowledge.

Furthermore, we introduce a feature enhancement mechanism that deepens the network's understanding of existing features, significantly improving the classification performance of the model while introducing new classes. We also propose sparse loss to optimize the model, reducing memory usage while better receiving new class knowledge. We validated our proposed framework on two publicly available incremental learning datasets. The empirical results

show that our method is superior to state-of-the-art methods, effectively reducing the forgetting of old class knowledge and improving the learning ability of new class knowledge.

2. Related Works

The research of incremental learning has made great progress, but how to mitigate catastrophic forgetting and maintain significant learning ability is still the focus of research. Most of the mainstream incremental learning methods alleviate catastrophic forgetting from the following four aspects: parametric regularization method, data playback method, class prototype playback method and network structure-based method.

Parametric regularization methods reduce the forgetting of old knowledge by setting additional constraints on the updating of model parameters. Kirkpatrick et al. [4] first proposed to reduce catastrophic forgetting by constraining important parameters with Elastic weight consolidation (EWC). Specifically, the method uses the Frisher information matrix to calculate importance. The Synaptic intelligence (SI) [5] method calculates the importance of each parameter in the model online during the training phase. MAS(Memory aware synapses) [6] method uses unlabeled samples to estimate the sensitivity of this nonlinear mapping function of neural network to parameter changes, and acts as the parameter importance estimation. Although the above methods achieve good results in some task incremental learning, they generally perform poorly in class incremental learning Settings.

The data playback based approach is in incremental learning, allowing the model to hold a small number of old classes These small portions of the old class data will be used in conjunction with the new class data for current model updates. Rebuffi et al. [3] proposed the Incremental classifier and representation learning (iCaRL) based on data playback for the first time. After learning each task, the method samples each class and samples the most representative sample in the limited memory size for subsequent model training. In addition, there are incremental learning methods based on generated data playback that primarily utilize GAN[7] or conditional GAN [8] to generate pseudo-samples for old classes. The classification model is trained simultaneously with the generation model at each incremental stage. The FearNet approach proposed by Kemker et al. [9] uses auto-encoders [10] to consolidate old class knowledge: For each old class, FearNet uses an auto-encoder to generate a sample of the old class based on the class mean of the depth feature space, and then trains the current model with a real sample of the new class.

Yu et al. [11] proposed that for each old class, only the class prototype in the feature space should be saved for the joint classification of the old and new classes. In the face of the prototype drift problem caused by the update of feature extractor, literature [11] proposed a Semantic drift compensation (SDC) strategy. According to the amount of feature deviation of the current task data in the new and old feature space, the drift of the old class prototype is approximated, and the SDC will map the saved old class prototype to the new location in the feature space. PASS (Prototype augmentation and self-supervision) [12] only keeps the mean value of newly learned classes in the depth feature space as the prototype. When learning new classes, simple prototype enhancement based on Gaussian noise can overcome the imbalance between old and new weights in the classifier. This kind of method can maintain strict incremental Settings, and at the same time, it has better anti-forgetting performance.

By extending the network structure, Liu et al. [13] proposed a new type of Adaptive aggregation networks (AANets) to explicitly solve the stability-plasticity problem in class incremental learning. Yan et al. [14] decoupage the feature representation from the classifier [15], and on this basis, a dynamically expanded table is proposed

The Dynamically expandable representation (DER) method, which constructs a modular deep classification network consisting of a network of super feature extractors and linear classifiers. In the test phase, the feature extractor used to map the test sample is the model after the final parameter fusion, and the feature space of the old class has also changed. Therefore, the structure expansion strategy mentioned above cannot guarantee that the old class knowledge will not be forgotten in theory.

3. Methods

In this section we propose a solution to the class incremental learning problem, which aims to mitigate the catastrophic forgetting problem and enable better learning of new classes. To this end, we propose a dynamic structure expansion strategy to avoid forgetting, and we also apply feature enhancement methods to incremental learning. In order to maintain the performance of the classes seen earlier, a sparsity regularization loss strategy is also used. In categorical incremental learning, the model is required to learn from a constantly updated stream of data. In the incremental step $t \in [1..T]$, let Y_t be the set of the new class and D_t be the data set containing the sample (x, y), where x is the input image and $y \in Y_t$ is the corresponding label. In order to maximize the classification accuracy, the complete classification before step t can be made and a good prediction can be made, that is, $Y_{[1:t]} = \bigcup_{i=1}^{t} Y_i$.

3.1. Dynamic Structure Extension

Our model architecture is built on the basis of dynamic structure extension, and its core consists of three main parts: a shared primary feature extraction layer L, a dynamically extensible set of high-level feature extractors, and an integrated classifier.

The shared primary feature extraction layer L remains unchanged throughout model training and is configured with its parameter set θ_L. Its role is to capture primary features of the input data that serve as the basis for subsequent processing. Because it is shared, it ensures that the primary feature representations that have been learned before can be utilized when the model learns a new class.

We progressively develop a dynamically scalable collection of advanced feature extractors tailored to learning requirements. Each new advanced feature extractor H_s maintains the same structure but possesses an independent parameter set θ_{H_s} As new classes or tasks are introduced, additional H_t is incorporated into the collection.

During incremental learning, we fix the old parameter set $\{\theta_L, \theta_{H_1}, ..., \theta_{H_{t-1}}\}$, and encourage the new advanced feature

extractor H_t to exclusively acquire knowledge related to the new class and produce the d-dimensional vector h_s.

The extensible unified classifier F_t is an independent layer with weights are made up of the matrix W_t. Where each column vector of W_t corresponds to each class $i \in Y_{[1:t]}$. These weight vectors are not completely new, but are extended based on the weight matrix W_{t-1} in the previous step $t-1$. Specifically, the expansion process consists of two directions: column direction expansion to include features from the newly added advanced feature extractor H_t, and row direction expansion to cover the newly added class Y_t. Let W_{t-1} be the matrix of weight vectors $w_{t-1,i}$ of all previously known classes $i \in Y_{[1:t]}$. Construct a new weight matrix W_t, which can be expressed as $W_t = [[W_{t-1} \circ V_t]; U_t]$.Where V_t is a feature weight matrix from H_1 to H_{t-1}, and for the new class in Y_t, U_t is a matrix whose weight vector corresponds to the H_t features of all classes in $Y_{[1:t]}$.If $z_t = [h_1 \circ h_2 \circ \cdots \circ h_t]$ is the join of the feature vectors obtained from the input image x after passing through all advanced feature extractors, then the probability of the input x prediction belonging to class i can be estimated as follows:

$$p_i(z_t) = \frac{e^{\eta \cdot sim(z_t, w_{t,i})}}{\sum_{j \in Y_{[1:t]}} e^{\eta \cdot sim(z_t, w_{t,i})}} \quad (1)$$

where η is a learnable scalar and $sim(\cdot)$ is the cosine similarity between two vectors?

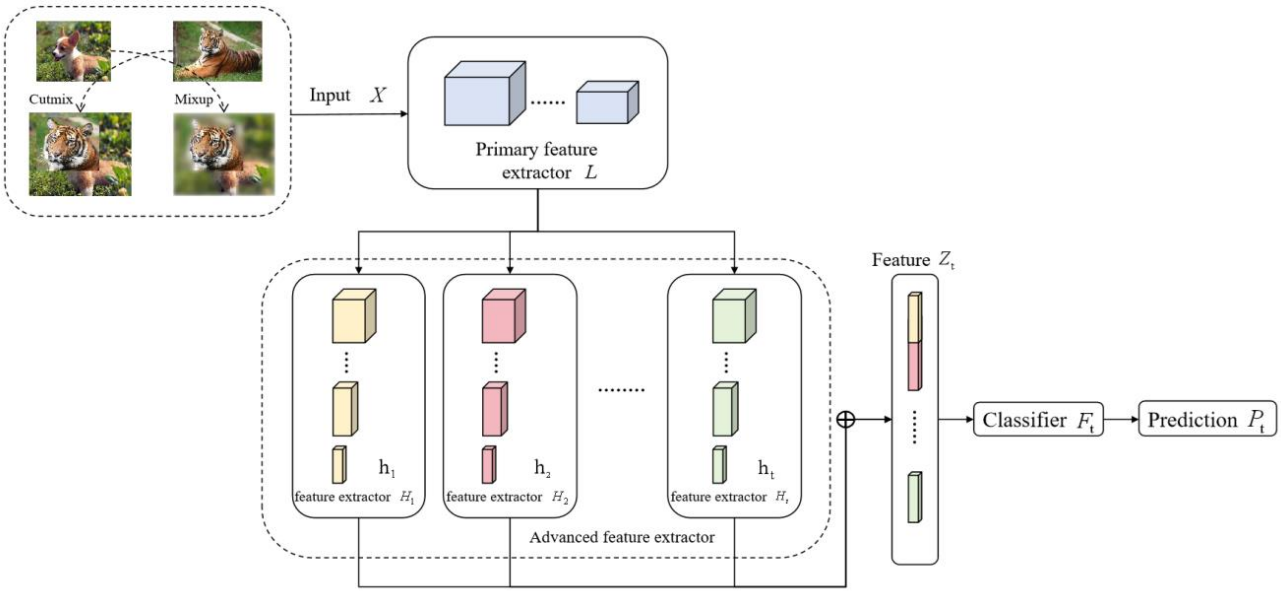

Figure 2. Model architecture based on dynamic structure extension and feature enhancement

3.2. Feature enhancement

Drawing on the valuable experience of previous research, we innovatively adopted a data enhancement strategy where we cleverly incorporated two technologies, Mixup[16] and CutMix[17], to achieve diverse data enhancement goals.

We apply parallel fusion operation and use Mixup technology to mix the data and labels of different samples through linear interpolation to generate new training samples.

$$x_1 = mixup(x_i, x_j) = \lambda x_i + (1-\lambda) x_j \quad (2)$$

Where $\lambda \in [0,1]$ is a random coefficient that controls the proportion of two samples in the mixing process.

CutMix takes it a step further by cutting part of one sample directly and pasting it onto the other, while adjusting the labels of both samples according to the size of the cropped area.

$$x_2 = CutMix(x_i, x_j) = M \odot x_i + (1-M) \odot x_j \quad (3)$$

In our framework, we use parallel fusion operations to create challenging "derived class" samples, forcing models to learn more robust feature representations to cope with complex and variable data environments.

$$L_{ce} = \sum L_{ce}(y_i, f_\theta(x_i)), i = 1,2 \quad (4)$$

Although this "derived class" information is not directly used in the subsequent incremental process, it effectively promotes the adaptability of type $f(z)$ to previously unseen types.

3.3. Loss compensation

Our training sample R_t consists of the incoming data set D_t and a finite number of samples of each seen class in memory M_{t-1} at each incremental step t > 1.Our goal is to optimize the new model parameters θ_{H_t}, V_t, U_t while maintaining the old parameters θ_L, θ_{H_1},..., $\theta_{H_{t-1}}$, w_{t-1} fixed to retain previous knowledge. Objective L focuses on learning the discriminant features of the new input data set,

which has four components: sparse loss, distillation loss and auxiliary loss:

$$L_{loss} = L_{Spar} + \lambda_1 L_{Dist} + \lambda_2 L_{Aux} \quad (5)$$

Where λ_1 and λ_2 are the hyper-parameters.

Sparsity Loss

Sparse regularization is an optimization technique that encourages the sparsity of a model by adding a regular term to the loss function. This regular term is often related to the weight of the model, and at each step in the incremental process, we encourage the model to minimize the number of parameters with minimal performance degradation. Inspired by this, we add sparsity losses based on the proportion of used weights among all available weights.

$$L_{Spar} = -\sum_{i=1}^{n} p_i(z_{t-1}) \log(p_i(z_t)) + \lambda \sum_{i=1}^{n} |w_i| \quad (6)$$

where n is the number of samples in R_t with class label Y .Where λ is the hyper-parameters

Distillation Loss

We need to freeze previously learned features in our scheme, and to mitigate catastrophic forgetting, we add a regularization term designed to transfer knowledge from the old model to the new model. We therefore use the Logits stage distillation loss (Rebuffi et al. 2017) [3] by distillation to minimize the Kullback-Leibler divergence between the probabilities of the old class knowledge $Y_{[1:t-1]}$ predicted by the model in the previous step as follows:

$$L_{Dist} = \mathop{\mathrm{E}}_{(x,y)\sim S_t} \left[\left| Y_{[1:t-1]} \right| \sum_{i \in Y_{[1:t-1]}} p_i(z_{t-1}) \log \frac{p_i(z_{t-1})}{p_i(z_t)} \right] \quad (7)$$

Considering that the need to freeze and preserve previously learned knowledge varies with the number of old classes, we weight the losses.

Auxiliary loss

In order to learn the discriminant features of the new class, we introduce an auxiliary loss. It uses class-balanced focal loss, but focuses only on the features extracted by H_t ,that is h_t and the weight vector $u_{t,y}$ corresponding to class y as follows?

$$L_{Aux} = \mathop{\mathrm{E}}_{(x,y)\sim R_t} \left[-\frac{1-\delta}{1-\delta^n} \left(1 - p_y(h_t)\right)^{\gamma} \log\left(p_y(h_t)\right) \right] \quad (8)$$

Where

$$p_i(h_t) = \frac{e^{\eta \cdot sim(h_t, w_{t,i})}}{\sum_{j \in Y[1:t]} e^{\eta \cdot sim(h_t, w_{t,i})}} \quad (9)$$

Therefore, in the classifier F_t we use, existing parameters can learn better decision boundaries in new feature dimensions.

4. Experiments and Results

In this section, we will conduct a number of experiments to verify the effectiveness of the algorithm. In particular, we evaluated our approach using a widely used bench marking protocol on the CIFAR-100[18], ImageNet-100[19] datasets. We also conducted a series of ablation studies to assess the importance of each component and provide additional insights into our approach. Below, we first introduce the experimental setup and implementation details in Section 4.1, and then introduce the benchmarks and evaluation indicators of our comparative experiments in Section 4.2. We then present the evaluation results of the two datasets in Section 4.3. Finally, we introduce the ablation study and analysis for our method in Sec. 4.4.

4.1. Experimental setup

4.1.1. Datasets and Settings

We used the following data sets and Settings in our experiment: CIFAR-100 [18] consists of 32 × 32-pixel color images of 100 classes. It contains 50,000 images for training, 500 images per class, and 10,000 images for assessment, 100 images per class. ImageNet-1000 [19] is a massive data set of 1000 classes, including approximately 1.2 million RGB images for training and 50,000 images for validation. ImageNet-100 [19] is built by selecting 100 classes from the ImageNet-1000 data set. The protocol we follow is that we use half of the classes to train the model and evenly split the remaining classes for training at each incremental step (Hou et al. 2019) [20]. If the sample has a total of 100 classes, start with a model trained with 50 classes, and the remaining 50 classes are divided into 5 and 10 steps, each class has 20 samples as memory. Based on previous work, we used a group selection strategy to select a fixed number of 20 samples for each class in all experiments for data playback.

4.1.2. Implementation Details

All of our methods are implemented in PyTorch and trained on an NVIDIA 3090 with 32GB of memory. For ImageNet100, we use ResNet-18 (He et al. 2016) [21] as the backbone of the network and cosine normalization (Hou et al. 2019) [20] at the classifier layer. For CIFAR100, we use ResNet32 as the backbone network, and for our method, we use the first two residual blocks as primary feature extractors and introduce copies of the remaining two residual blocks at each incremental step. The backbone of all models was initialized with pre-trained weights on ImageNet (Deng et al. 2009) [19] and optimized with an SGD optimizer with a value attenuation coefficient of 0.0005 and a momentum value of 0.9. The batch size for the model training phase of CIFAR-100 and ImageNet-100 is 32.

4.1.3. Baselines

We compared our approach to the following baselines :(a) iCaRL (Rebuffi et al. 2017) [3] uses logits stage distillation loss to mitigate forgetting; (b) LUCIR (Hou Saihui et al. 2019) [20] solves problems such as unbalanced classifier weight and knowledge bias of old classes by using strategies such as cosine normalization, smaller forgetting constraints and isolation between classes ; (c) The LwF (Zhi zhong Li et al.) [22] method, drawing on model distillation techniques, mitigated the forgetting problem to some extent by retaining the predictions of the old model on the new data as soft labels, which were used together with the labels of the new task to train the new model ; (d) DER (Yan, Xie, and He 2021) [14] uses dynamic extensible representations to handle new classes without forgetting them, and two-stage learning to solve class imbalances.

4.1.4. Evaluation Metrics

An intuitive evaluation index is to investigate after each

incremental step t, only evaluate the classification accuracy of all known classes after the training of all classes seen so far, and record it as reflecting the overall performance of the model in the data flow learning process. The model accuracy rate of step t can be averaged to obtain the average accuracy rate of the model in Y_i:

$$Ac = \frac{1}{T}\sum_{t=1}^{T}\left[\frac{1}{t}\sum_{i=1}^{t}A_t^i\right] \tag{10}$$

In addition to Ac as a benchmark, we also quantified the amount of forgetting of the old class as the difference between the accuracy of the current step and the previously obtained maximum, the amount of decline in classification accuracy on task i ,based on these forgetting degree $A_t^i = \max_{j\in\{1,\cdots,t-1\}}(A_j^i - A_t^i)$,the average forgetting rate of the incremental learning phase is calculated :

$$Fg = \frac{1}{T}\sum_{t=1}^{T}\left[\frac{1}{t-1}\sum_{i=1}^{t-1}A_t^i\right] \tag{11}$$

4.2. Comparative Study

Table 1 summarizes the results of the ImageNet-100 and CIFAR-100 datasets. There are also two types of incremental settings defined for each data set, resulting in 4 columns of comparison results. We saw that our proposed approach outperformed all baselines in increments of 5 and 10 steps. At the same time, compared with the mainstream dynamic structure extension method, the proposed method also has obvious advantages. This fully shows that the method combined with feature data enhancement strategy can identify new classes better on the premise of limited knowledge of old classes.

Specifically, compared with the second best method DER, the performance of the proposed method on the CIFAR-100 datasets is improved by 1.3% and 1.6% at 5 and 10 steps, respectively. The advantage on the ImageNet-100 datasets is a further 2.8% and2.3%.

Under the two types of incremental settings of the CIFAR-100 data set, we respectively show its accuracy at each incremental step, as shown in Figure 3and Figure 4.

Table 1. Accuracy results for CIFAR-100 and ImageNet-100 datasets

Accuracy/%	CIFAR-100		ImageNet-100	
Metric	5 steps	10 steps	5 steps	10 steps
LwF	48.1	45.6	62.6	58.6
LUCIR	56.6	50.7	58.4	52.9
iCaRL	65.1	61.2	68.8	64.6
DER	75.3	73.1	77.8	74.4
Ours	77.6	74.8	80.6	76.7

We observe that DER and our proposed method have the most similar data curve effects, demonstrating the effectiveness of dynamic extended representations in preserving old features. But thanks to the two-stage learning approach used in DER, where the classifier is re-initialized and fine-tuned in the second stage using a balanced data set. Discarding previously learned class weight vectors corresponding to old features leads to a decrease in accuracy and an increase in forgetting. And our average accuracy curve is significantly better than DER.

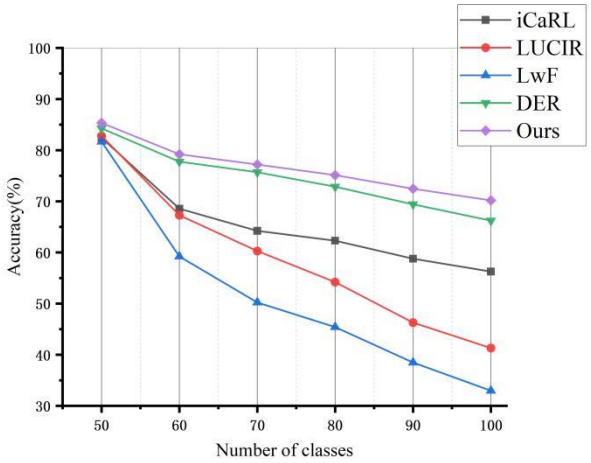

Figure 3. Experimental results on CIFAR-100(5 incremental steps)

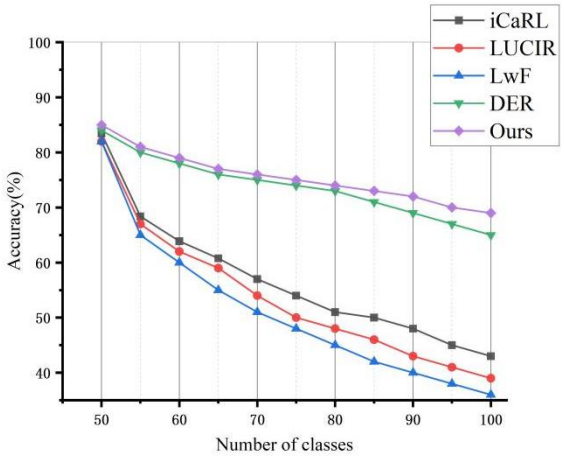

Figure 4. Experimental results on CIFAR-100(10incremental steps)

Under the two types of incremental settings of the ImageNet-100 data set, we respectively show its accuracy at each incremental step, as shown in Figure 5 and Figure 6.

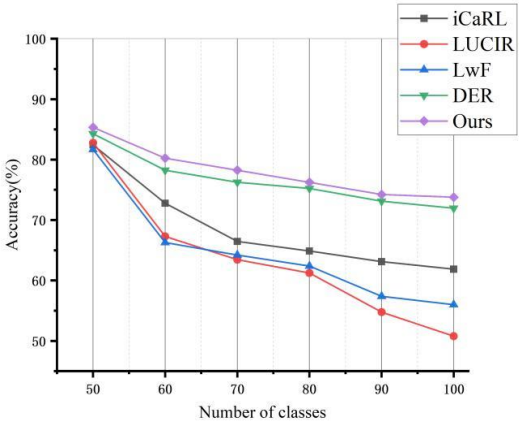

Figure 5. Experimental results on ImageNet-100(5 incremental steps)

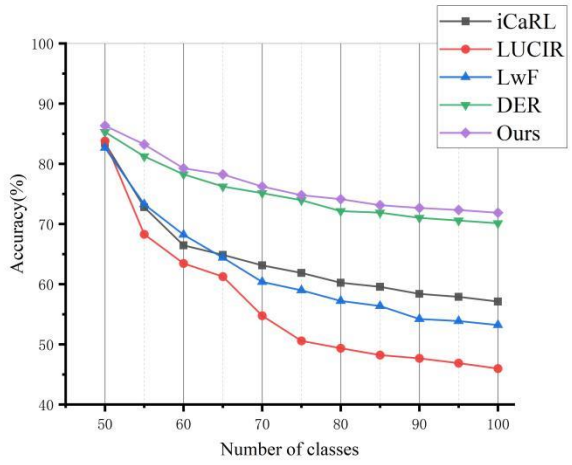

Figure 6. Experimental results on ImageNet-100(10incremental steps)

4.3. Ablation Study and Analysis

We conducted exhaustive ablation studies to assess the contribution of each component to our approach. Since all the loss components are incorporated, our method achieves the highest Ac,. We conducted ablation experiments in the multi-type incremental scenarios of the CIFAR-100 and lmageNet-100 datasets, and the corresponding experimental results are listed in Table 2. It can be found that the dynamic structure extension model combined with the feature enhancement strategy achieves the optimal classification accuracy in all experimental scenarios.

Table 2. Ablation study on ClFAR-100 and lmageNet-100 datasets

Accuracy/%	CIFAR-100		ImageNet-100	
Metric	5 steps	10 steps	5 steps	10 steps
ALL	77.6	74.8	80.6	76.7
Without L_{Spar}	72.2	68.9	77.6	74.8
Without L_{Dist}	74.5	69.2	78.1	73.3
Without L_{Aux}	75.3	69.8	77.3	74.6

In the 5 steps of lmageNet-100, compared with the base model, the improvement is 3.0%, 2.5% and 3.3%, .This is strong evidence that our model can effectively enhance the ability to learn new classes.

5. Conclusion

In this paper, we propose a new class incremental learning method, which utilizes dynamic scalable representations to achieve more efficient class incremental learning, while retaining previously learned features and adapting new features, effectively reducing the case of catastrophic forgetting. In addition, by incorporating two powerful data enhancement techniques, Mixup and CutMix, into our framework, we designed a feature enhancement mechanism that significantly improves the classification performance of the model while introducing new classes. This scheme ensures efficient identification of old and new classes without compromising the efficiency of previous models, so that new discriminant features can be better learned. We conduct detailed experiments on two main incremental classification benchmarks. The experimental results show that our method is consistently superior to other methods.

References

[1] Xu M, Guo L Z, “Learning from group supervision: The impact of supervision deficiency on multi-label learning,” Science China Information Sciences,2021, vol. 64, pp.1–13.

[2] Lippi M, Montemurro M A, Degli Esposti M, et al. Natural language statistical features of LSTM-generated texts. lEEE Transactions on Neural Networks and Learning Systems, 2019, pp.3326-3337.

[3] Rebuffi S A, Kolesnikov A, Sperl G, Lampert C H. iCaRL: Incremental classifier and representation learning. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, USA: IEEE, 2017, pp.5533−5542.

[4] Kirkpatrick J, Pascanu R, Rabinowitz N, Veness J,Desjardins G, Rusu A A, et al. Overcoming catastrophic forgetting in neural networks. Proceedings of the National Academy of Sciences of the United States of America, 2017, vol. 144, no. 13, pp.3521−3526 .

[5] Zenke F, Poole B, Ganguli S. Continual learning through synaptic intelligence. Proceedings of the 34th International Conference on Machine Learning (ICML).Sydney, Australia:PMLR, 2017,pp.3987−3995.

[6] Aljundi R, Babiloni F, Elhoseiny M, Rohrbach M, TuytelaarsT. Memory aware synapses: Learning what (not) to forget. Proceedings of the 15th European Conference on Computer Vision (ECCV). Munich, Germany: Springer, 2018, pp. 144−161.

[7] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, et al. Generative adversarial networks.Communications of the ACM, 2020,vol. 63, no. 11,pp.139−144.

[8] Odena A, Olah C, Shlens J. Conditional image synthesis with auxiliary classifier GANs. Proceedings of the 34th International Conference on Machine Learning (ICML). Sydney, Australia: PMLR, 2017, pp.2642−2651.

[9] Kemker R, Kanan C. FearNet: Brain-inspired model for incremental learning. Proceedings of the 6th International Conf-erence on Learning Representations (ICLR). Vancouver,Canada: OpenReview.net, 2018.

[10] Kingma D P,Welling M. An introduction to variational au-toencoders. Foundations and Trends Foundations and Trendsin Machine Learning in Machine Learning,2019,vol. 12, no. 4,pp.307-392.

[11] Yu L,Twardowski B, Liu XL, Herranz L,Wang K, Cheng YM, et al.Semantic driftcompensation for class-incremental learning. Proceedings of the IEEE/CVF Conference onComputer Vision and Pattern Recognition(CVPR). Seattle.USA:IEEE,2020,pp.6980-6989.

[12] Zhu F, Zhang X Y, Wang C, Yin F, Liu CL. Prototype augmentation and self-supervision for incremental learning. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(OVPR). Nashville, USA: IEEE,2021,pp.5867-5876.

[13] Liu Y Y, Schiele B,Sun QR. Adaptive aggregation networks for class-incremental learning. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR).Nashville,USA:IEEE,2021,pp.2544-2553.

[14] Yan S P,Xie JW,He X M. DER: Dynamically expandable representation for class incremental learning. Proceedings of

the IEEE/OVF Conference on Computer Vision and Pattern Recognition(CVPR).Nashville,USA:IEEE,2021,pp.3014-3023.

[15] Kang B Y, Xie S N, Rohrbach M, Yan Z C, Gordo A, Feng JS,et al. Decoupling representation and classifier for long-tailed recognition.In:Proceedings of the 8th International Conference on Learning Representations Ethiopia:Open-Review.net, 2020.

[16] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin et al.Mixup: Beyond Empirical Risk Minimization.International Conference on Learning Representations 2018.

[17] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, et al.CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features.IEEE International Conference on Computer Vision, 2019 .

[18] Krizhevsky A, Hinton G. Learning multiple layers of features from tiny images. Handbook of Systemic Autoimmune Diseases, 2009, vol. 1, no.4,pp. 41−60 .

[19] Deng J, Dong W, Socher R, Li L J, Li K, Li F F. ImageNet: A large-scale hierarchical image database. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Miami, USA: IEEE, 2009, pp. 248−255.

[20] Hou, S.; Pan, X.; Loy, C. C.; Wang, Z.; and Lin, D. 2019. Learning a unified classifier incrementally via rebalancing. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp.831–839

[21] He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp.770–778.

[22] Li Z Z, Hoiem D. Learning without forgetting. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, vol. 40, no.12,pp.2935−2947.