Scientific Research Publishing

Open Journal of Applied Sciences, 2022, 12, 421-433
https://www.scirp.org/journal/ojapps
ISSN Online: 2165-3925
ISSN Print: 2165-3917

Dual Channel with Involution for Long-Tailed Visual Recognition

Mengxue Li

School of Sciences, Hebei University of Technology, Tianjin, China
Email: 1287447348@qq.com

How to cite this paper: Li, M.X. (2022) Dual Channel with Involution for Long-Tailed Visual Recognition. *Open Journal of Applied Sciences*, **12**, 421-433.
https://doi.org/10.4236/ojapps.2022.124029

Received: February 23, 2022
Accepted: April 4, 2022
Published: April 7, 2022

Abstract

With the rapid increase of large-scale problems, the distribution of real-world datasets tends to be long-tailed. Existing solutions typically involve re-balancing strategies (*i.e.*, re-sampling and re-weighting). Although they can significantly promote the classifier learning of deep networks, they will unexpectedly impair the representative ability of the learned deep features to a certain extent. Therefore, this paper proposes a dual-channel learning algorithm with involution neural networks (DC-Invo) to take care of representation learning and classifier learning concurrently. In this work, the most important thing is to combine ResNet and involution to obtain higher classification accuracy because of involution's wider coverage in the spatial dimension. The paper conducted extensive experiments on several benchmark vision tasks including Cifar-LT, Imagenet-LT, and Places-LT, showing that DC-Invo is able to achieve significant performance gained on long-tailed datasets.

Keywords

Long-Tailed Recognition, Deep Neural Network, Dual-Channel Structure, Involution

1. Introduction

Visual recognition research has developed rapidly during the past few years, mainly driven by large image datasets [1] [2], deep convolutional neural networks (CNNs) and high-performance computing resources. In the traditional classification and recognition tasks, the distribution of training data is often artificially balanced. Visual phenomena, however, are more data biased. In the form of long-tailed distribution [3] [4], many standard methods fail to model correctly, resulting in a significant decrease in accuracy. Motivated by this, there have been some recent attempts to study long-tailed recognition, *i.e.*, recognition in

environments where the number of instances in each class is highly variable and follows a long-tailed distribution.

When learning with long-tailed datasets, a common challenge is that instance-rich (or heads) classes dominate the training process. The learned classification model performs better on these classes, however, it performs significantly worse for instance-scare (or tail) classes. To solve this problem and to improve the performance of all classes, prominent and effective approach is the class re-balancing strategy, which is proposed to mitigate the extreme imbalance of training data. In general, class re-balancing methods can be roughly divided into two groups, *i.e.*, re-sampling [5] [6] and re-weighting [7] [8]. These methods can adjust network training by re-sampling instances or re-weighting the losses of samples within the SGD mini-batches, which are expected to be closer to the test distribution. Therefore, class re-balancing can effectively directly affect the classifier weights' update of the deep network, *i.e.*, promoting classifier learning.

However, although re-balancing methods have good ultimate predictions, these methods still have adverse effects, *i.e.*, they can also unexpectedly impair the representativeness of the learned deep features (*i.e.*, representation learning) to some extent. Specifically, when the data imbalance is extreme, there are risks of over-fitting the tail data (by over-sampling) and under-fitting the whole data distribution (by under-sampling). For re-weighting, it distorts the original distribution by directly changing or even reversing the frequency of data presentation. To solve these problems, the BBN model [9] proposed a unified bilateral branch network to carry out feature learning and classifier learning of deep network simultaneously and a cumulative learning strategy to adjust bilateral learning for exhaustively improving the recognition performance of long-tailed tasks.

Moreover, convolution has been a central component of modern neural networks, triggering the explosion of deep learning in vision. In 2021, Li *et al.* [10] reconsidered the inherent principles of standard convolution for visual tasks, especially spatial-agnostic and channel-specific. Instead, they proposed a new neural network operator by inverting the above design principles of convolution, named involution. More specifically, involution kernels are distinct in the spatial extent but shared across channels. Involution can summarize context in a broader spatial arrangement, thus overcoming the difficulties of modeling long-range interactions well, and can adaptively assign weights in different locations to prioritize visual elements with the most information in the spatial domain.

Based on the above, this paper proposes a dual-channel structure with involution neural networks (DC-Invo) for both representation learning and classifier learning. At the same time, combined with DC-Invo model training, the cumulative learning strategy is used to adjust bilateral learning. As shown in **Figure 1**, the DC-Invo model consists of two channels, called the "traditional learning channel" and the "re-balancing learning channel". As the name implies, the traditional learning channel adopts uniform sampling to maintain the original data

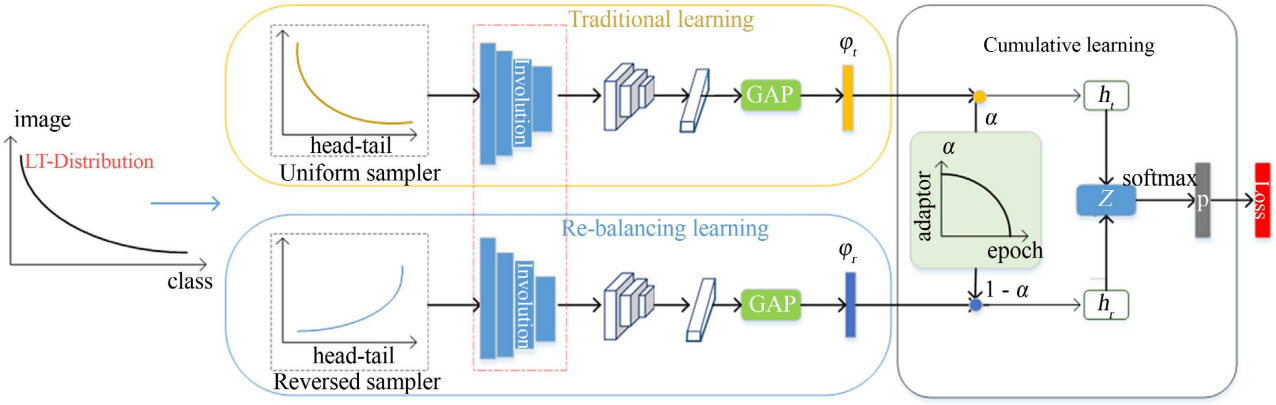

Figure 1. Framework of our DC-Invo.

distribution structure for representation learning. While, the re-balancing learning channel used a reversed sampler (*i.e.*, small sampling weights for high frequency samples) to model the tail data. The predicted outputs of these dual channels are then aggregated in the cumulative learning part by an adaptive trade-off parameter *α*. *α* is automatically generated by the "Adapter" based on the number of training epochs, which adjusts the entire DC-Invo model to firstly learn general features from the original distribution and then gradually focus on tail data. More importantly, in the backbone network model, the involution neural network is combined with ResNet residual network to obtain higher classification accuracy on the long-tailed datasets because of the involution kernel's wider coverage in the spatial dimension (wider receptive field).

To demonstrate the effectiveness of the proposed DC-Invo, the paper conducts extensive experiments on four benchmark long-tailed datasets: CIFAR-10-LT, CIFAR-100-LT, Imagenet-LT, Places-LT. Empirical results on these datasets show that the model obviously outperforms existing state-of-the-art methods.

Summarily, the primary contributions of this paper are as follows: 1) The paper proposed a dual-channel learning algorithm with involution neural networks (DC-Invo) to deal with representation learning and classifier learning for exhaustively enhancing long-tailed recognition. In addition, a cumulative learning strategy is used to adjust bilateral learning. 2) The paper evaluated the DC-Invo model on four benchmark long-tailed visual recognition datasets, achieving higher accuracy than established state-of-the-art methods (different sampling strategies and new loss designs).

2. Relaxed Work

2.1. Re-Sampling

Re-sampling is a preprocessing technique to solve the problem of imbalanced data classification. In the past, a large number of sampling techniques have been proposed from different perspectives, mainly oversampling by simply repeating data for minority classes [11] [12] [13] and under-sampling by abandoning data for dominant classes [14] [15] However, re-sampling is not a really perfect solu-

tion because the tail data are often learned repeatedly, which lacks enough sample differences and is not robust enough, and the head data is often not fully learned [16] [17].

2.2. Cost-Sensitive Learning

The cost-sensitive function is an effective method to deal with unbalanced classification, which is mainly to make the model pay more attention to the few samples in the learning process, so as to alleviate the phenomenon that the model is too biased towards the majority of samples. Cost-sensitive function methods mainly include the adjustment of sample weights, the design of various types of loss functions, and techniques that are beneficial to the learning of a few types of samples. Ren *et al.* [18] proposed an approach based on primary learning, which automatically assigned weights to the training set samples according to the loss of validation set. In terms of the loss function, various novel loss functions have emerged in recent years. In 2017, Lin *et al.* [19] designed Focal Loss, a Loss function for online mining of difficult samples. In 2018, Dong *et al.* [20] added a kind of corrected loss on the basis of the Softmax loss function. Cui *et al.* [21] designed a weight adjustment scheme, which used the effective sample number of each class to adjust the weight of class loss, so as to generate a class balanced loss function. Cao *et al.* [22] proposed the LDAM (Label-Distor-Aare Margins) loss function, which encourages the decision boundary of model learning to be as far away from a few classes as possible, and theoretically and rigorously proved the rationality of the loss function.

3. Methodology

As shown in **Figure 1**, our DC-Invo mainly adds a new neural network operator to the backbone network structure of the BBN model [9], including three main components: traditional learning channel, re-balancing learning channel and cumulative learning strategy. The traditional learning channel obtains the input data from a uniform sampler, which is responsible for learning the general patterns of the original distribution. While the re-balancing channel receives input data from a reversed sampler and is designed to model tail data. The cumulative learning strategy aggregates output feature vectors φ_t and φ_r of the two channels to calculate the training loss.

3.1. Involution

Involution is a new neural network operator proposed by Li *et al.* in 2021 [11], which inverted the two inherent principles of convolution: spatial-agnostic into spatial-specific, and channel-specific into channel-agnostic. Finally, based on the two design principles (*i.e.*, spatial-specific and channel-agnostic), a new type of operator was proposed, called involution. Compared with convolution, involution can aggregate the context in a wider space so as to overcome the difficulty of modeling remote interactions well and can adaptively allocate the weights of

different positions so as to prioritize the visual elements with the most abundant information in the spatial domain.

Let $X \in R^{H\times W\times C}$ denote the input feature map, where H, W represent its height, width and C enumerates the channels. The kernel of involution is $H\times W\times K\times K\times G$, where $G \ll C$, indicates that all channels share G kernels. So the involution can be formulated as:

$$Y_{i,j,k} = \sum_{u,v\in\Delta_k} H_{i,j,u+\lfloor K/2\rfloor,v+\lfloor K/2\rfloor,\lfloor kG/C\rfloor} X_{i+u,j+v,k} \tag{1}$$

where $H \in R^{H\times W\times K\times K\times G}$ is involution kernel.

The general form of involution kernel generation is as follows:

$$H_{i,j} = \phi\left(X_{\Psi_{i,j}}\right) \tag{2}$$

where $\Psi_{i,j}$ is an index set of the neighborhood of (i, j), therefore, $X_{\Psi_{i,j}}$ represents a patch containing $X_{i,j}$ in the feature map.

The paper [11] proposed a simple and effective instantiation of the kernel generating function ϕ. $\Psi_{i,j}$ is the set of points $\{(i, j)\}$, *i.e.*, $X_{\Psi_{i,j}}$ is taken as a single pixel with (i, j) in the feature map, then the instance of the generation of the involution kernel is obtained:

$$H_{i,j} = \varphi\left(X_{i,j}\right) = W_1\sigma\left(W_0 X_{i,j}\right) \tag{3}$$

where $W_0 \in R^{\frac{C}{r}\times C}$ and $W_1 \in R^{(K\times K\times G)\times\frac{C}{r}}$ represent linear transformation matrix, γ represents reduction ratio and σ implies Batch Normalization and non-linear activation functions that interleave two linear projections.

As shown in **Figure 2**, under the above simple instantiation of involution kernel, a complete schematic diagram of involution can be obtained.

The schematic is from the literature [11]. For the feature vector on a point of the input feature map, it is first expanded into the shape of the kernel through ϕ (FC-BN-ReLU-FC) and reshape (channel-to-space) transformation to obtain the corresponding involvement kernel on this coordinate point, and then

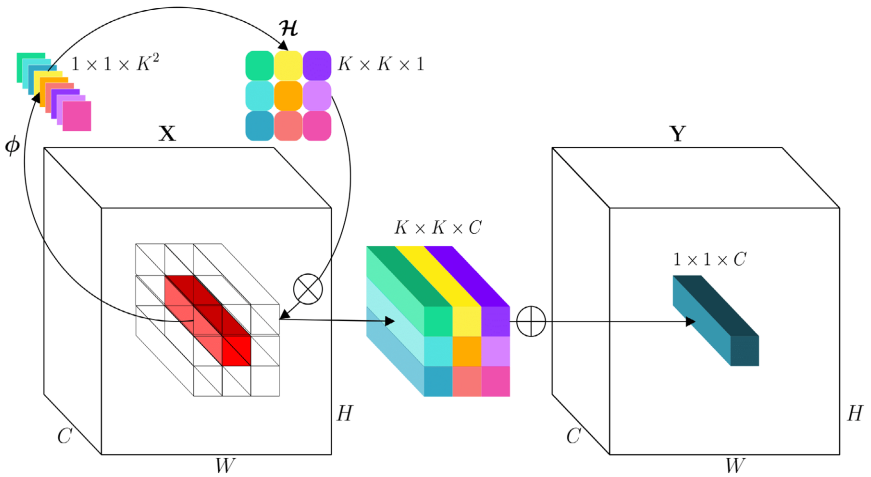

Figure 2. Simple instance generation diagram of involution.

multiply-add with the feature vector in the neighborhood of this coordinate point on the input feature map to obtain the final output feature map.

3.2. Modeling Process

Let $D=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$ denote a long-tailed distribution dataset containing C categories, where N is the number of samples. Assuming that n_i represents the number of samples of the ith category, then $N=\sum_{i+1}^{N} n_i$. In general, the subscripts of categories are sorted in descending order by the number of samples. If $i<j$, then $n_i>n_j$ and $n_1 \gg n_C$.

In the data input stage, the traditional learning channel adopts uniform samplers to maintain the original data distribution and obtain input data (x_t,y_t). And the re-balancing learning channel adopts an inverted sampler to model tail data (the sample is sampled inversely according to the frequency of the sample, *i.e.*, the high frequency sample has a smaller weight) to acquire input data (x_r,y_r). Then, two samples are fed into their corresponding channels to obtain the feature vectors φ_t and φ_r. Next, the weights of φ_t and φ_r are controlled by adaptive trade-off parameters α, and the weighted feature vectors $\alpha\varphi_t$ and $(1-\alpha)\varphi_r$ are sent to classifier h_t and h_r respectively. The output will be integrated by element-wise addition, and the results are as follows:

$$Z=\alpha h_t^{\mathrm{T}}\varphi_t+(1-\alpha)h_r^{\mathrm{T}}\varphi_r \tag{4}$$

At this point, Z is the predicted output, and then the softmax function is used to normalize Z to get the probability of each class:

$$\hat{p}_i=\frac{\mathrm{e}^{z_i}}{\sum_{j=1}^{C}\mathrm{e}^{z_j}} \tag{5}$$

The weighted distribution cross-entropy classification loss of our DC-Invo model is illustrated as:

$$L=\alpha L_t+(1-\alpha)L_r \tag{6}$$

where $L_t=-\sum_{i=1}^{n_t} y_i^t \log(\hat{p}_i)$ and $L_r=-\sum_{i=1}^{n_r} y_i^r \log(\hat{p}_i)$ are cross-emtropy loss function of each channel.

3.3. Proposed Cumulative Learning Strategy

A cumulative learning strategy is proposed to dynamically adjust the learning focus between dual channels by controlling the feature weight generated by two channels and classification loss L. It is designed to learn the general patterns firstly, and then pay attention to the tail data gradually. In the training phase, the feature φ_t of the traditional learning channel will be multiplied by α and the feature φ_r of the re-balancing learning channel will be multiplied by $1-\alpha$, where α is automatically generated according to the training epoch. Assuming that the total number of training epochs of the model is expressed as $T_{\max}$, and the current epoch is expressed as T, the trade-off parameter α can be calculated as:

$$\alpha = 1 - \left(\frac{T}{T_{\max}} \right)^2 \tag{7}$$

With the increase of training epochs, α will be gradually decreased. The motivation is to make the learning focus of our DC-Invo should gradually change from feature representation to classifiers, which can significantly improve the accuracy of long-tailed recognition.

In the experiment, we also provide this intuitive result by comparing different types of adapters, cf. Section 4.4.3.

4. Experiments

4.1. Datasets and Empirical Settings

Long-tailed CIFAR-10 and CIFAR-100. According to the number of categories, CIFAR can be divided into CIFAR10 and CIFAR100 that contain 10 categories and 100 categories respectively. The two datasets respectively contain 60,000 images, 50,000 for training and 10,000 for validation. The paper generated the long-tailed version of CIFAR-10 and CIFAR-100 following those used in [22] with controllable degrees of data imbalance. The test dataset remains unchanged, and the number of samples of each category in the training dataset is set according to $n = n_i \cdot \mu^{\frac{i}{100}}$, where n_i is the original number of the class i, and μ is a long-tailed factor to describe the severity of the long-tail problem, e.g., $\mu = \frac{N_{\max}}{N_{\min}}$, Long-tailed factors the paper used in experiments are 20, 50 and 100.

Long-tailed Imagenet. The paper constructed the long-tailed version of Imagenet following those used in [23]. The validation set and test set remain unchanged, and each type of sample in the training set is sampled following the Pareto distribution, where the power value $\alpha = 6$. A total of 115,846 images were collected from the training dataset, with each category containing 1280 images at most and 5 images at least.

Long-tailed Places. The structure of the long-tailed version of Places is similar to that of Imagenet-LT. Following the settings in [23], 20 images are sampled from each category of the validation set, 50 images from each category of the test set, and samples from each category of the training set are sampled following the Pareto distribution with the power value $\alpha = 6$. The training data set obtained by sampled has a total of 62,500 images, with each category containing 4980 images at most and 5 images at least.

4.2. Implementation Details

Implementation details on CIFAR. For long-tailed CIFAR-10 and CIFAR-100, the paper followed the simple data augmentation proposed in [24] for training: a 32 × 32 crop is sampled randomly from the original image or its horizontal flip with 4 pixels which are padded on each size. The paper trained the combination of ResNet-32 [24] and involution as our backbone network and used the stan-

dard mini-batch stochastic gradient descent (SGD) with a momentum of 0.9, weight decay of 2×10^{-4} for all experiments. The paper trained all the models on a GeForce RTX 2080Ti GPU with a batch size of 128 for epochs. For a fair comparison, the initial learning rate is set to 0.1 and decayed by 0.01 at the 120th epoch and again at the 160th epoch for our DC-Invo. A linear warm-up learning rate schedule [25] is used for the first 5 epochs.

Implementation details on Imagenet-LT and Places-LT. For Imagenet-LT and Places-LT, all images are first adjusted to 256×256. During training, images are randomly cropped to 224×224, and then flip horizontally with a 50% probability. The paper used the standard mini-batch stochastic gradient descent (SGD) with a momentum of 0.9 to train 60 epochs, and the learning rate is initialized to 0.1, which decayed to 10% of the original at the 20th and 40th epochs, respectively.

4.3. Comparison Methods

In experiments, this paper compared DC-Invo model with several methods:

Focal Loss: A loss function, based on the Softmax cross-entropy loss function, increases the weight of difficult samples while reducing the weight of easy samples.

CB Loss: A weight adjustment scheme is designed to re-balance the losses using valid samples from each class.

LDAM Loss: By encouraging model learning, the decision boundaries are as far away from a few classes as possible.

OLTR: A knowledge transfer method, which solves the problem of insufficient feature representation due to the small number of tail category samples by maintaining a feature representation that enhances neural network learning in a visual memory bank.

BBN: A bilateral-branch network structure, which uses the original dataset for training on one side and the resampled balanced dataset for training on the other side. The learning of long-tailed data is improved by decoupling the feature learning and classifier learning.

4.4. Main Results

4.4.1. Experiment Result on Long-Tailed CIFAR

Table 1 reports the classification accuracy results of various long-tailed CIFAR-10 and CIFAR-100 datasets with three long-tailed factors: 20, 50, 100. This paper consistently demonstrates that DC-Invo achieves the best results on all datasets when compared with other methods, including Focal Loss, CB Loss, LDAM Loss and BBN. Additionally, it can be found from the table, when the long-tail factor is larger, the accuracy difference between DC-invo and other algorithms is larger. Especially, compared with BBN, DC-Invo has better classification accuracy improve that combination with ResNet and involution can improve the classification accuracy.

4.4.2. Experiment Result on Imagenet-LT and Places-LT

Table 2 shows the experimental results of different algorithms on Imagenet-LT and Places-LT datasets. Similar to the results on the CIFAR-LT dataset, the DC-Invo model outperformed other algorithms, for example, the Classification accuracy of Imagenet-LT and Places-LT is 2.1% and 1.3% higher than the second place algorithm, respectively.

In conclusion, the comprehensive comparison of different algorithms on several datasets shows that DC-Invo model can well model long-tail distributed datasets.

4.4.3. Different Cumulative Learning Strategies

To verify the effectiveness of the proposed cumulative learning strategy, we explore a number of different strategies to generate the adaptive trade-off parameter α on CIFAR-10-IR50. The abscissa represents the completion degree of model training, the ordinate represents the value of α used in the training period, and each curve presents how α varies with the training process of the model, cf. Figure 3. The paper tested with both progress relevant strategies which adjust α with the number of training epochs (*i.e.*, parabolic increment, cosine decay and linear decay, etc) and irrelevant strategies (*i.e.*, equal weight, single weight, and β-distribution), cf. Table 3.

Table 1. Top 1 accuracy for long-tailed CIFAR-10/100.

Dataset	Long-tailed CIFAR-10			Long-tailed CIFAR-100		
Imbalanced ratio	100	50	20	100	50	20
CE	66.3	71.4	80.2	36.3	38.9	52.0
Focal Loss	66.4	73.3	80.4	36.4	39.4	51.9
CB Loss	71.1	74.8	80.6	37.6	41.4	52.6
LDAM loss	69.8	76.5	82.9	39.4	43.6	53.1
OTLR	71.4	77.5	83.6	38.3	43.8	53.4
BBN	73.2	79.0	84.2	40.3	44.1	53.6
DC-Invo	77.7	80.6	84.9	43.6	46.1	53.9

Table 2. Top 1 accuracy for long-tailed imagenet and places.

Dataset	Long-tailed Imagenet	Long-tailed Places
CE	29.7	22.9
Focal Loss	30.5	23.5
CB Loss	35.8	26.4
LDAM loss	36.3	24.7
OTLR	35.6	25.4
BBN	37.7	26.1
DC-Invo	39.8	27.4

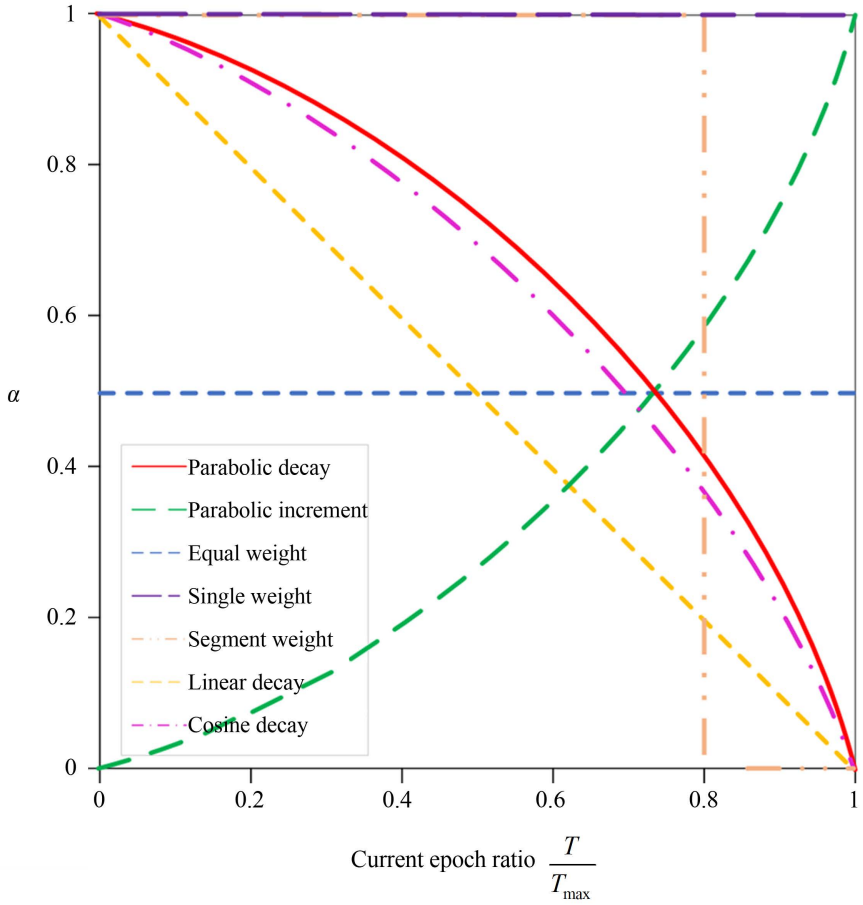

Figure 3. Schematic diagram of α generated by different course learning strategies.

Table 3. Ablation studies of different adaptor strategies of DC-Invo on Long-tailed CIFAR-10-IR-50.

Adaptor	α	Accuracy
Parabolic increment	$\left(\frac{T}{T_{\max}}\right)^2$	70.52
β-distribution	Beta (0.2, 0.2)	77.13
Equal weight	0.5	77.93
Single weight	1	78.62
Segment weight	$\frac{T}{T_{\max}}<1\rightarrow 1:0$	79.46
Linear decay	$1-\frac{T}{T_{\max}}$	79.29
Cosine decay	$\cos\left(\frac{T}{T_{\max}}\cdot\frac{\pi}{2}\right)$	79.09
Parabolic decay	$1-\left(\frac{T}{T_{\max}}\right)^2$	80.68

As shown in Table 3, the parabolic decay adapter is the best among these adapters. The results of the three decay strategies are all better than the single-weight strategy using a traditional learning channel. The results of the two channels using equal weight all the time are slightly lower than the single-weight strategy, and the parabolic increment strategy and the randomly generated β-distribution strategy have the worst results. These phenomena indicate that the model should emphasize representation learning first and then classifier learning. At the same time, compared with segment weight, the parabolic decay does not directly step from 1 to 0, but gradually decreases, so that the two channels can maintain the learning state simultaneously during the whole training process and the model pays attention to the tail data at the end of the iteration without damaging the learned features.

5. Conclusion

For long-tailed problems, some literature reveals class re-balancing strategies can not only promote classifier learning significantly but also damage representation learning to some extent. Motivated by this, this paper proposed a dual-channel structure with involution neural networks (DC-Invo) for both representation learning and classifier learning to effectively improve the recognition performance of long-tailed classification tasks. Through comparison with state-of-the-art methods and extensive ablation studies, this paper verified that our DC-Invo could achieve the best results on long-tailed benchmarks.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

[1] Jia, D., Wei, D., Socher R., Li, L.-J., Li, K. and Li, F-F. (2009) Imagenet: A Large-Scale Hierarchical Image Database. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (*CVPR* 2009), Miami, 20-25 June 2009, 248-255.

[2] Zhou, B.L., Khosla, A., Lapedriza, A., Torralba, A. and Oliva, A. (2016) Places: An Image Database for Deep Scene Understanding. https://doi.org/10.1167/17.10.296

[3] Wang, Y.-X., Ramanan, D. and Hebert, M. (2017) Learning to Model the Tail. *Advances in Neural Information Processing Systems* 30 (*NIPS* 2017), 7029-7039.

[4] Van Horn, G. and Perona, P. (2017) The Devil Is in the Tails: Fine-Grained Classification in the Wild.

[5] Chawla, N.V., Bowyer, K.W., Hall, L.O. and Kegelmeyer, W.P. (2002) Smote: Synthetic Minority Oversampling Technique. *Journal of Artificial Intelligence Research*, **16**, 321-357. https://doi.org/10.1613/jair.953

[6] Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. (2005) Borderline-Smote: A New Over-Sampling Method in Imbalanced Data Sets Learning. *Advances in Intelligent Computing*, *International Conference on Intelligent Computing*, Hefei, 23-26 August 2005, 878-887. https://doi.org/10.1007/11538059_91

[7] Li, B.Y., Liu, Y. and Wang, X.G. (2018) Gradient Harmonized Single-Stage Detector. *Proceedings of the AAAI Conference on Artificial Intelligence*, Honolulu, 27 January-1 February 2019, 8577-8584. https://doi.org/10.1609/aaai.v33i01.33018577

[8] Van Hulse, J., Khoshgoftaar, T.M. and Napolitano, A. (2007) Experimental Perspectives on Learning from Imbalanced Data. *Proceedings of the* 24*th International Conference on Machine Learning*, Corvallis, June 20-24 2007, 935-942. https://doi.org/10.1145/1273496.1273614

[9] Zhou, B.Y., *et al.* (2019) BBN: Bilateral-Branch Network with Cumulative Learning for Long-Tailed Visual Recognition. 2020 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Seattle, 13-19 June 2020, 9716-9725. https://doi.org/10.1109/CVPR42600.2020.00974

[10] Li, D., Hu, J., Wang, C., *et al.* (2021) Involution: Inverting the Inherence of Convolution for Visual Recognition, 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, 20-25 June 2021, 12316-12325. https://doi.org/10.1109/CVPR46437.2021.01214

[11] Chawla, N.V., Lazarevic, A., Hall, L.O., *et al.* (2003) SMOTEBoost: Improving Prediction of the Minority Class in Boosting. 7*th European Conference on Principles of Data Mining and Knowledge Discovery*, Dubrovnik, 22-26 September 2003, 107-119. https://doi.org/10.1007/978-3-540-39804-2_12

[12] He, H., Bai, Y., Garcia, E.A., *et al.* (2008) ADASYN: Adaptive Synthetic Sampling Approach for Imbalanced Learning. 2008 *IEEE International Joint Conference on Neural Networks* (*IEEE World Congress on Computational Intelligence*), Hong Kong, 1-8 June 2008, 1322-1328.

[13] Maciejewski, T. and Stefanowski, J. (2011) Local Neighbourhood Extension of SMOTE for Mining Imbalanced Data. 2011 *IEEE Symposium on Computational Intelligence and Data Mining* (*CIDM*), Paris, 11-15 April 2011, 104-111. https://doi.org/10.1109/CIDM.2011.5949434

[14] Kubat, M., Matwin, S. (1997) Addressing the Curse of Imbalanced Training Sets: One-Sided Selection. *Proceedings of the* 14*th International Conference on Machine Learning*, San Francisco, 179-186.

[15] Yen, S.J. and Lee, Y.S. (2009) Cluster Based Under-Sampling Approaches for Imbalanced Data Distributions. *Expert Systems with Applications*, **36**, 5718-5727. https://doi.org/10.1016/j.eswa.2008.06.108

[16] Chen, H., Li, Y.N., Chen, C.L. and Tang, X.O. (2016) Learning Deep Representation for Imbalanced Classification. 2016 *IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), Las Vegas, 27-30 June 2016, pages 5375-5384.

[17] Japkowicz, N. and Stephen, S. (2002) The Class Imbalance Problem: A Systematic Study. *Intelligent Data Analysis*, **6**, 429-449. https://doi.org/10.3233/IDA-2002-6504

[18] Ren, M.Y., Zeng, W.Y., Yang, B. and Urtasun, R. (2018) Learning to Reweight Examples for Robust Deep Learning. *Proceedings of the* 35*th International Conference on Machine Learning*, Stockholm, 4334-4343.

[19] Lin, T.-Y., Goyal, P., Girshick, R., *et al.* (2017) Focal Loss for Dense Object Detection. 2017 *IEEE International Conference on Computer Vision* (*ICCV*), Venice, 22-29 October 2017, 2999-3007. https://doi.org/10.1109/ICCV.2017.324

[20] Dong, Q., Gong, S. and Zhu, X. (2019) Imbalanced Deep Learning by Minority Class Incremental Rectification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **41**, 1367-1381. https://doi.org/10.1109/TPAMI.2018.2832629

[21] Cui, Y., Jia, M.L., Lin, T.-Y., Song, Y. and Belongie, S. (2019) Class-Balanced Loss Based on Effective Number of Samples. 2019 *IEEE/CVF Conference on Computer*

Vision and Pattern Recognition (*CVPR*), Long Beach, Long Beach, 9268-9277. https://doi.org/10.1109/CVPR.2019.00949

[22] Cao, K.D., Wei, C.L., Gaidon, A., Arechiga, N. and Ma, T.Y. (2019) Learning Imbalanced Datasets with Label Distribution-Aware Margin Loss. *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Vancouver, 1567-1578.

[23] Liu, Z.W., *et al.* (2019) Large-Scale Long-Tailed Recognition in an Open World. 2019 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Long Beach, 15-20 June 2019, 2532-2541. https://doi.org/10.1109/CVPR.2019.00264

[24] He, K.M., Zhang, X.Y., Ren, S.Q. and Sun, J. (2016) Deep Residual Learning for Image Recognition. 2016 *IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), Las Vegas, 27-30 June 2016, 770-778. https://doi.org/10.1109/CVPR.2016.90

[25] Goyal, P., Doll'ar, P., Girshick, R., *et al.* (2017) Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour.