IMPROVING CLASS BALANCING AT BOTH FEATURE EXTRACTOR AND CLASSIFIER HEAD

Kanghao Chen, Huijuan Lu*, Ruixuan Wang† and Wei-Shi Zheng*

School of Computer Science and Engineering, Sun Yat-sen University, China;
chenkh25@mail2.sysu.edu.cn, luhj6@mail2.sysu.edu.cn,
†wangruix5@mail.sysu.edu.cn, wszheng@ieee.org.

2022 IEEE International Conference on Multimedia and Expo (ICME) | 978-1-6654-8563-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICME52920.2022.9860019

ABSTRACT

Training data are often imbalanced across classes in practice, and such class imbalance issue often causes model predictions biased toward majority classes during inference. Different from existing solutions which employ various training strategies to alleviate the class imbalance issue, this study proposes a novel two-head model architecture to help alleviate the issue. One auxiliary classifier head helps the feature extractor of the classifier more fairly learn to extract features for each class, and the main classifier head learns in a more class-balanced manner by dividing each majority class into multiple clusters in advance and considering each cluster as a new class. Extensive empirical evaluations on four class-imbalanced image datasets showed that the proposed approach achieves state-of-the-art classification performance.

Index Terms— class imbalance, feature balancing, class division.

1. INTRODUCTION

Deep neural networks have shown their superior performance on various image classification tasks [1–3]. However, when training samples are imbalanced across classes as widely observed in real scenarios [4], the trained classifiers often have biased predictions toward majority classes having larger training samples, and the minority classes having smaller training samples are often ignored to some extent in both classifier training and inference [5, 6].

Multiple approaches have been proposed to handle such class imbalance issue. One group of approaches try to re-balance classes during model training, e.g., by re-sampling to obtain a similar number of training data for each class [7, 8], or by class re-weighting in the training loss [9–11]. With the class re-balancing strategies, the classifier would handle the majority classes and minority classes more fairly during model training. However, class re-balancing based on a very limited number of training samples from the minority classes often causes model over-fitting for the minority classes. Since over-fitting corresponds to less generalizability of classifiers, another group of approaches try to directly improve the generalizability of classifiers, e.g., by transfer learning with a pre-trained classifier backbone using large dataset ImageNet [3], or by augmenting the number of training data particularly for minority classes with various augmentation techniques like Mixup [12] and its extensions Remix [13] and Balanced-Mixup [14]. Besides augmentation in the data space, augmentation of minority classes in the feature space also helps alleviate the over-fitting issue [15, 16]. Obviously, the above two groups of approaches can be combined to handle the class imbalance issue. For example, the state-of-the-art two-stage training strategy [17] firstly trains a more generalizable feature extractor, and then fine-tunes the classifier head with re-balancing strategies. The two-stage strategy is further extended to a cumulative learning strategy BBN [18], where the first feature extractor learning is smoothly shifted to the class re-balancing process. Another example is MiSLAS [19] which combines class re-weighting and Mixup augmentation during classifier training. All the existing approaches alleviate the class imbalance issue by improving the training process. In contrast, we propose a novel model architecture that can directly help alleviate the class imbalance issue.

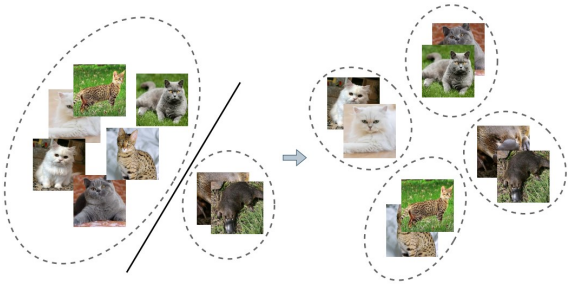

Fig. 1. Two demonstrative classes from an image classification dataset. There are many more images for the majority class 'cat' than for the minority class 'platypus'. This study divides the majority class into multiple clusters to alleviate the class imbalance issue during model training.

*Equal contribution.
†Corresponding author.

Considering that a convolutional neural network (CNN) classifier is often composed of a feature extractor and a following classifier head, we propose a novel two-head model structure to respectively help the feature extractor fairly handle all the classes and help the classifier head train in a class-balanced manner. As an auxiliary head branch, the first classifier head separates the feature map outputs of the feature extractor into multiple groups, with each group containing an equivalent number of feature maps and corresponding to a unique class, and enforces each group of feature maps to be responsible for the prediction of one specific class. In this way, the feature extractor would be trained to handle each class fairly with a small subset of output feature channels, regardless of the number of training samples for each class. Such fair handling of each class is expected to help the feature extractor learn to extract features unbiasedly for each class, and the class-specific group of feature channels also helps the prediction of each (particularly minority) class to be less affected by the other (particularly majority) classes. On the other hand, to avoid training a class-imbalanced classifier, the training samples of each majority class are firstly divided into multiple clusters, and each cluster is considered as a new class for the main classifier head (Figure 1). In this way, training samples are more balanced across (new) classes for the main classifier head, and such class-balanced training would largely reduce the negative effect (e.g., larger weight magnitude for majority classes [19]) of class imbalance on the classifier head. Extensive evaluations on four class-imbalanced datasets show that the proposed framework achieves state-of-the-art classification performance, supporting the effectiveness of the proposed framework. The main contributions of this study are summarized below.

- A novel auxiliary classifier head is proposed to help the feature extractor fairly handle each class.
- A class-division strategy is proposed to help train the classifier head in a more class-balanced manner, largely alleviating the negative impact of class imbalance.
- The classifiers trained with the proposed framework achieved state-of-the-art performance on four representative class-imbalanced classification tasks.

2. METHODOLOGY

In order to alleviate the class imbalance issue, we propose a novel framework to help the classifier more fairly learn from each class (Figure 2). To reduce the bias of the feature extractor toward majority classes, a feature balancing strategy is proposed such that the same amount of feature components from the output of the feature extractor is responsible for each class (Figure 2, feature-balancing head). To reduce the bias of the classifier head toward majority classes, a class division strategy is proposed (Figure 2, class-division head), i.e., the training data of each majority class are divided into multiple clusters, and each cluster is considered as one separate (new) class. In this way, the training dataset becomes class-balanced during model training.

2.1. Feature balancing

Considering that re-balancing strategies from the model input (e.g., re-sampling) and the model output (e.g., class re-weighting) are effective in alleviating the class imbalance issue, we argue that re-balancing inside the model may also help. With this consideration, we propose a feature balancing strategy based on an auxiliary task for model training (Figure 2, left part with green background), particularly by enforcing that a similar number of feature channels from the output of the feature extractor are responsible for each class.

Formally, given a training image $\mathbf{x}_i$ and the corresponding ground-truth class label $y_i \in \{1, 2, ..., C\}$, where C is the total number of classes, denote by $\mathbf{F}_i \in \mathbb{R}^{D\times H\times W}$ the feature map output from the feature extractor of the convolutional neural network backbone (e.g., ResNet-50), with width W, height H and channel number D. Then, the feature maps $\mathbf{F}_i$ are divided into C equivalent groups (Figure 2, right part for an example), with each group containing D/C (an integer without loss of generality) feature channels. The c-th group of feature channels $\mathbf{F}_{ic}$ is then convolved with a class-specific kernel $\mathbf{K}_c$, followed by a global average pooling (GAP), i.e.,

$$z_{ic} = GAP(\mathbf{F}_{ic} * \mathbf{K}_c)\,, \tag{1}$$

where $*$ denotes the convolution operator. Note that the collection of the class-specific convolutions over all the classes is actually a special group convolution, with a single convolution for each group. Denote by $\mathbf{z}_i = (z_{i1}, z_{i2}, \ldots, z_{iC})^{\mathrm{T}}$ the vector containing all z_{ic}'s, and denote by $g(\cdot)$ the softmax function, then the feature balancing strategy can be implemented by adding a regularization term $\mathcal{L}_1$ during training,

$$\mathcal{L}_1 = \frac{1}{N}\sum_{i=1}^{N} l_{CE}(g(\mathbf{z}_i), y_i)\,, \tag{2}$$

where l_{CE} is the cross-entropy loss and N is the total number of training data. By minimizing this loss term, a small and equivalent number of unique feature channels are enforced to represent the visual features relevant to each class. This would probably help ensure fairness in feature learning across all the classes, thus largely preventing the feature extractor from mainly learning features of the majority classes [19].

2.2. Class division

Class imbalance often causes significantly larger magnitude of model weights associated with outputs of majority classes in the classifier head [19]. Such consequent imbalance in model weights in turn leads to biased classification prediction

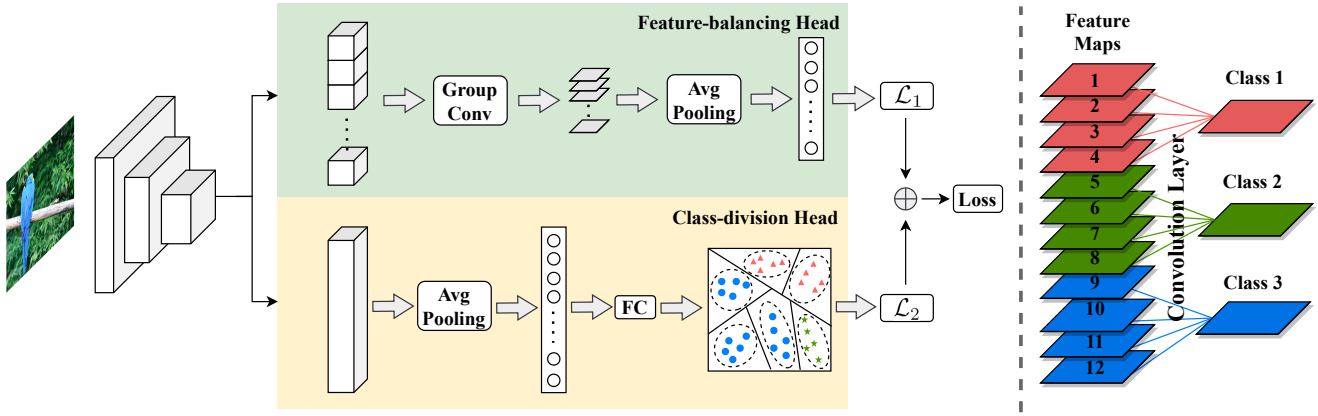

Fig. 2. The proposed learning framework to alleviate the class imbalance issue. The feature-balancing head (with green background) is proposed to help the feature extractor more fairly learn from each class, and the class-division head (with yellow background) is proposed to directly alleviate the class imbalance issued by training a class-balanced classifier. Right side of the dashed line: the demonstrative single-layer group convolution in the feature-balancing head.

toward majority classes during inference. To alleviate such weight imbalance, remedy strategies include directly reducing the magnitude of weights associated with majority classes or fine-tuning the classifier head using class re-balancing strategies at a post-processing stage [17, 20]. Different from these remedy strategies, a class-division strategy is proposed here such that a class-balanced classifier is trained, therefore fundamentally avoiding the weight imbalance in the classifier head. The basic idea is to divide the training samples of majority classes into multiple clusters and consider each cluster as a new class for model training.

Suppose the classes are re-ordered increasingly based on the number of training samples, and denote by n_c the number of training samples for the re-ordered c-th class (so $n_c \leq n_{c+1}$). If the training samples of the largest class are divided into a predefined u number of clusters, and the smallest class corresponds to a single cluster, then the c-th class can be divided into u_c clusters based on the linear relationship between the training samples of these classes, i.e.,

$$u_c = 1 + \left[\frac{n_c - n_1}{n_C - n_1}(u-1) \right], \tag{3}$$

where $[\cdot]$ represents the rounding operator. Various clustering strategies can be adopted to generate clusters for each class. Totally $K = \sum_{c=1}^{C} u_c$ clusters would be obtained, and the classifier head is designed to output K new classes (rather than the original C classes), i.e., a K-class classifier would be trained (Figure 2, the classifier head with yellow background).

Specially, considering that the training samples belonging to the same original class but different clusters often contain more similar visual features than the samples from different original classes, smoothed labels are designed for those training samples belonging to multiple clusters (i.e., correspondingly new classes) but from the same original class. Specifically, if one training sample belongs to one of the u_c (where $u_c > 1$) clusters from the original c-th class and the cluster index in the totally K clusters is denoted by k, then a predefined higher soft ground-truth value α (e.g., 0.5) is set to the k-th cluster (i.e., k-th new class) and a non-zero soft label value $(1-\alpha)/(u_c - 1)$ is set to each of the other $(u_c - 1)$ clusters belonging to the same original class. With the soft labels for training samples, the K-class classifier can be trained partly by minimizing the cross-entropy loss $\mathcal{L}_2$,

$$\mathcal{L}_2 = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} q_{ik} \log p_{ik}\,, \tag{4}$$

where p_{ik} is the k-th output of the classifier (with softmax activation at the output layer), and q_{ik} is the k-th element in the soft label vector for the i-th training sample. Together with the feature-balancing loss term $\mathcal{L}_1$, the whole system can be jointly trained by minimizing the overall loss $\mathcal{L} = \mathcal{L}_2 + \lambda\mathcal{L}_1$, where λ is the coefficient constant to trade off the two loss terms. Once the classifier is well trained, for any test data, the K-class classifier is employed to predict the class of the data with the help of the correspondence between the K clusters and the original C classes.

3. EXPERIMENTS

3.1. Experimental setting

Datasets: The proposed approach was evaluated based on the CIFAR-10, the CIFAR-100 and the 200-class Tiny-ImageNet [21] with controllable degrees of class imbalance, as well as a large-scale imbalanced dataset ImageNet-LT [4]. Following the evaluation of LDAM [20], on the first three datasets, class-imbalanced dataset versions were created by exponentially reducing the number of training examples

across classes while keeping the test set unchanged. The imbalance ratio β between the sample sizes of the largest class and the smallest class is used to describe the degree of class imbalance. Following the previous studies [18, 19], β was respectively set to 10, 50 and 100. Imagenet-LT [4] contains 1000 classes of totally 115.8K images, with class sample sizes ranging from 5 to 1280.

Implementation details: Following previous studies, ResNet-32, ResNet-18, and ResNeXt-50 were adopted as classifier backbones respectively for CIFAR, Tiny-ImageNet, and ImageNet-LT datasets. The SGD optimizer with momentum 0.9 was used to train classifiers for all experiments, with batch size 128, 128, and 512 respectively, and training over 200, 200, and 90 epochs respectively on CIFAR, Tiny-ImageNet, and ImageNet-LT datasets. On two CIFAR datasets and Tiny-ImageNet, the initial learning rate was 0.1 and decayed by 0.01 at the 120th and 160th epoch respectively. On ImageNet-LT, the initial learning rate was 0.2 and decayed by a cosine schedule [22]. The widely used data augmentations (e.g., random crop and flipping) were used for all model training. The cluster number u was set to 10 by default for all experiments, and random splitting was used to divide a larger class into multiple clusters considering that the proposed approach is insensitive to clustering strategies (see Section 3.3). The coefficient λ was set 1.5 for all experiments.

Baseline methods: The proposed approach was compared with multiple baseline methods, including not only the basic cross-entropy loss (CE), focal loss (Focal) [23], class balance (CB) [11], deferred re-sampling (DRS) [20] and Mixup [12] data augmentation, but also the recently proposed hybrid strategies CB+Focal [11], LDAM+DRW [20], BBN [18], MiSLAS [19] and Hybrid-SC [24]. For the large-scale dataset ImageNet-LT, OLTR [4] and LWS [17] were also used for comparison. The suggested hyper-parameter settings from the original studies were used in our own implementation.

3.2. Effectiveness evaluation

Evaluations on CIFAR: Two groups of comparisons were performed on both CIFAR-10 and CIFAR-100 datasets with the imbalance ratios 10, 50, and 100 respectively. In the first group (Table 1, first six rows), the proposed approach was compared with those methods each of which only employs one type of strategy. In the second group (Table 1, last seven rows), considering that the state-of-the-art approaches often employ two or more types of strategies, our approach also combines with Mixup and class re-weighting [11] (in Table 1, last row) as used in MiSLAS [19]. From the first group of results, it can be observed that the proposed approach consistently outperforms all the single-strategy baselines on both CIFAR datasets with various imbalance ratios. In the second group, the proposed approach achieves state-of-the-art performance on CIFAR-10 and clearly outperforms all the recently proposed methods (e.g., MiSLAS, Hybrid-SC) on CIFAR-100. Figure 3 shows that the proposed approach achieves significant performance gain mainly on those minority classes and more balanced weight norms over classes in the classifier head compared to those of the CE baseline, confirming that the proposed approach can help alleviate the class imbalance issue by improving the performance on minority classes.

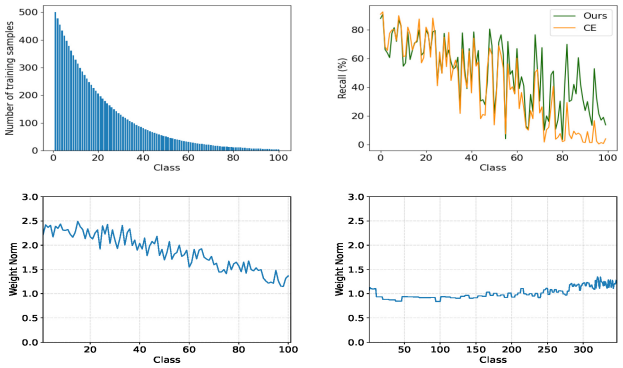

Fig. 3. Performance on CIFAR-100 with imbalance ratio 100. Top left: number of training data over classes. Top right: recall performance over classes. Second row: weight norm of classifier head for each class based on the CE baseline (Left) and ours (Right). The classes were ordered with decreasing number of training samples along the x-axis.

Evaluations on ImageNet: Similar results were obtained on the Tiny-ImageNet and the ImageNet-LT datasets. On the Tiny-ImageNet with all the imbalance ratios, the proposed approach outperforms all the single-strategy baselines (Table 2, first five rows), and the combination of our approach with Mixup and class re-weighting outperforms all the state-of-the-art hybrid-strategy methods (Table 2, last five rows). On the ImageNet-LT with the preset class imbalance, the proposed approach again outperforms the single-strategy baselines (Table 3, first six rows, last column) and achieves state-of-the-art performance compared to hybrid-strategy methods (Table 3, last five rows, last column). When inspecting more detailed performance on the majority classes (with more than 100 training images per class), the medium-size classes (with 20 to 100 images per class), and the minority classes (less than 20 images per class) respectively, it can be observed that the proposed approach significantly improves the performance on both minority and medium-size classes compared to the basic training strategies CE, Focal, and Mixup, and meanwhile does not downgrade the performance on the majority classes (Table 3). All these results suggest that the proposed approach can effectively help alleviate the class imbalance issue.

3.3. Ablation and robustness study

Effect of two classifier heads: Table 4 shows that, with varying levels of imbalance ratios, both the feature-balancing head (FBH; second row) and the class-division head (CDH; third

Table 1. Performance (top-1 accuracy, %) comparison on the CIFAR-10 and CIFAR-100 datasets with varying imbalance ratios. RB: re-balancing; DA: data augmentation; ES: ensemble; RL: representation learning. Since test set is class-balanced, top-1 accuracy is equivalent to mean recall over all classes.

Group	Methods	Combined strategies	CIFAR-10			CIFAR-100		
			$\beta = 100$	$\beta = 50$	$\beta = 10$	$\beta = 100$	$\beta = 50$	$\beta = 10$
Single	CE	-	70.36	74.81	86.39	38.32	43.85	55.71
	Focal	RB	70.38	76.72	86.66	38.41	44.32	55.78
	Mixup	DA	73.06	77.82	87.10	39.54	44.99	58.02
	CB	RB	72.37	78.96	86.54	33.99	45.41	57.12
	LDAM	RB	73.35	78.40	86.54	39.42	44.75	56.27
	Ours	RB	**78.79**	**82.12**	**88.58**	**44.92**	**50.31**	**62.85**
Hybrid	CB+Focal	RB+RB	74.57	79.27	87.49	39.60	45.32	57.99
	DRS	RL+RB	75.61	79.81	87.38	41.61	45.48	58.11
	LDAM+DRW	RB+RB	77.03	81.03	88.16	42.04	46.62	58.71
	BBN	ES+RB	79.82	82.18	88.32	42.56	47.02	59.12
	MiSLAS	RL+RB+DA	82.10	**85.70**	90.00	47.00	52.30	63.20
	Hybrid-SC	ES+RB	81.40	85.36	**91.12**	46.72	51.87	63.05
	Ours-Full	RB+RL+DA	**82.46**	84.85	88.97	**50.08**	**55.70**	**66.60**

Table 2. Performance comparison on Tiny-ImageNet.

Methods	$\beta = 100$	$\beta = 50$	$\beta = 10$
CE	27.35	31.03	43.71
Focal	27.80	31.25	43.33
Mixup	29.30	33.61	46.42
LDAM	27.87	31.44	46.10
Ours	**36.28**	**40.62**	**52.98**
DRS	28.59	33.39	45.83
LDAM+DRW	33.57	37.30	49.81
BBN	34.47	38.49	47.16
MiSLAS	37.72	42.75	53.47
Ours-Full	**41.16**	**45.86**	**56.71**

Table 3. Performance comparison on Imagenet-LT.

Method	Majority	Medium	Minority	All
CE	65.9	37.5	7.7	44.4
Focal	64.3	37.1	8.2	43.7
Mixup	68.3	39.4	8.8	45.6
LDAM	65.0	42.2	15.7	46.7
OLTR	59.9	45.8	27.6	48.7
Ours	64.1	47.8	24.0	**50.3**
DRS	57.5	44.6	27.5	46.8
LDAM+DRW	60.4	46.8	28.8	49.1
LWS	60.2	47.2	30.3	49.9
MiSLAS	65.1	50.4	32.9	53.2
Ours-Full	66.4	50.6	31.0	**53.5**

row) respectively can help improve the classification performance compared to the conventional CNN classifier trained with the cross-entropy (first row). The combination of the two heads further improves the performance (last row), altogether supporting the effectiveness of both classifier heads in alleviating the class imbalance issue.

Table 4. Ablation study on CIFAR-100 and Tiny-ImageNet.

FBH	CDH	CIFAR-100/Tiny-ImageNet		
		$\beta = 100$	$\beta = 50$	$\beta = 10$
-	-	38.32/27.35	43.85/31.03	55.71/43.71
✓	-	43.07/33.01	48.49/36.62	61.61/49.12
-	✓	43.63/30.45	49.55/34.76	62.57/47.37
✓	✓	44.92/36.28	50.31/40.62	62.85/52.98

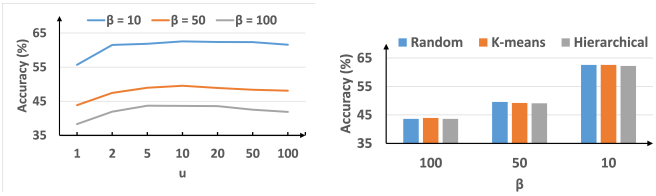

Fig. 4. Performance of the proposed approach on CIFAR-100 with different cluster number u (Left) and clustering strategies (Right).

Sensitivity of hyper-parameters: Figure 4 (Left) shows the class-division head itself (excluding the feature-balancing head) can help improve the performance when the cluster number u varies within a relatively large range (e.g., $[5, 20]$), and Figure 4 (Right) shows that the performance of the proposed approach changes little when different clustering strategies (Random splitting, K-means, Hierarchical clustering) were adopted to divide the training samples of larger class into multiple clusters. These results support that the proposed approach is insensitive to the choice of hyper-parameter values during model training.

On different backbones: As shown in Table 5, the proposed approach consistently outperforms existing methods on different model backbones, further supporting the robustness and generalizability of the proposed approach.

Table 5. Performance comparison on CIFAR-100 with different model backbone architectures. Imbalance ratio is 100.

Method	ResNet-110	VGG-16	MobileNet-v2
CE	42.47	40.25	37.11
Mixup	44.78	40.60	37.57
DRS	44.18	42.14	42.03
LDAM+DRW	43.38	40.22	43.55
MiSLAS	47.55	46.27	44.17
Ours-Full	**50.45**	**47.85**	**44.66**

4. CONCLUSION

In this paper, a novel two-head framework is proposed to help alleviate the class imbalance issue. The feature-balancing head can help the feature extractor handle each class fairly, while the class-division head divides each majority class into multiple new classes such that the classifier is trained in a more class-balanced manner. Experiments on four image classification datasets clearly support that the proposed approach can particularly improve the performance on minority classes. The combination of the proposed approach with existing strategies can further improve classification performance. We expect the proposed approach can be employed in more scenarios like object detection and image segmentation.

Acknowledgement. This work is supported by the NSFC programs (No. 62071502, U1811461), the Guangdong Key Research and Development Programs (No. 2020B1111190001, 2019B020228001), and the Meizhou Science and Technology Program (No. 2019A0102005).

5. REFERENCES

[1] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[2] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.

[3] Olga Russakovsky, Jia Deng, and Hao Su et al., "Imagenet large scale visual recognition challenge," *IJCV*, 2015.

[4] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X. Yu, "Large-scale long-tailed recognition in an open world," in *CVPR*, 2019.

[5] Nathalie Japkowicz and Shaju Stephen, "The class imbalance problem: A systematic study," *Intell. Data Anal.*, 2002.

[6] Haibo He and Edwardo A. Garcia, "Learning from imbalanced data," in *TKDE*, 2009.

[7] Li Shen, Zhouchen Lin, and Qingming Huang, "Relay backpropagation for effective learning of deep convolutional neural networks," in *ECCV*, 2016.

[8] Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural networks*, 2018.

[9] Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang, "Learning deep representation for imbalanced classification," in *CVPR*, 2016.

[10] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert, "Learning to model the tail," in *NeurIPS*, 2017.

[11] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge J. Belongie, "Class-balanced loss based on effective number of samples," in *CVPR*, 2019.

[12] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz, "mixup: Beyond empirical risk minimization," in *ICLR*, 2018.

[13] Hsin-Ping Chou, Shih-Chieh Chang, Jia-Yu Pan, Wei Wei, and Da-Cheng Juan, "Remix: Rebalanced mixup," in *ECCV Workshops*, 2020.

[14] Adrian Galdran, G. Carneiro, and Miguel Ángel González Ballester, "Balanced-mixup for highly imbalanced medical image classification," in *MICCAI*, 2021.

[15] Peng Chu, Xiao Bian, Shaopeng Liu, and Haibin Ling, "Feature space augmentation for long-tailed data," in *ECCV*, 2020.

[16] Yangwen Hu, Zhehao Zhong, Ruixuan Wang, Hongmei Liu, Zhijun Tan, and Wei-Shi Zheng, "Data augmentation in logit space for medical image classification with limited training data," in *MICCAI*, 2021.

[17] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis, "Decoupling representation and classifier for long-tailed recognition," in *ICLR*, 2020.

[18] Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen, "Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition," in *CVPR*, 2020.

[19] Zhisheng Zhong, Jiequan Cui, Shu Liu, and Jiaya Jia, "Improving calibration for long-tailed recognition," in *CVPR*, 2021.

[20] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Aréchiga, and Tengyu Ma, "Learning imbalanced datasets with label-distribution-aware margin loss," in *NeurIPS*, 2019.

[21] Ya Le and Xuan S. Yang, "Tiny imagenet visual recognition challenge," 2015.

[22] Ilya Loshchilov and Frank Hutter, "SGDR: stochastic gradient descent with warm restarts," in *ICLR*, 2017.

[23] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár, "Focal loss for dense object detection," *TPAMI*, 2020.

[24] Peng Wang, K. Han, Xiu-Shen Wei, Lei Zhang, and Lei Wang, "Contrastive learning based hybrid networks for long-tailed image classification," in *CVPR*, 2021.