LMK thesis

by LI Mengke

Submission date: 14-May-2022 02:45PM (UTC+0800)
Submission ID: 1804847853
File name: 93134_LI_Mengke_LMK_thesis_554854_997799875.pdf (17.02M)
Word count: 34050
Character count: 181673

Advances in Long-Tailed Visual Recognition

LI Mengke

A thesis submitted in partial fulfilment of the requirements

for the degree of

Doctor of Philosophy

Principal Supervisor:

Prof. CHEUNG Yiu-ming (Hong Kong Baptist University)

May 2022

DECLARATION

I hereby declare that this thesis represents my own work which has been done after registration for the degree of PhD at Hong Kong Baptist University, and has not been previously included in a thesis or dissertation submitted to this or any other institution for a degree, diploma or other qualifications.

I have read the University's current research ethics guidelines, and accept responsibility for the conduct of the procedures in accordance with the University's Research Ethics Committee (REC). I have attempted to identify all the risks related to this research that may arise in conducting this research, obtained the relevant ethical and/or safety approval (where applicable), and acknowledged my obligations and the rights of the participants.

Signature: _____

Date: May 2022

Abstract

Real-world data tends to have a long-tailed distribution. Existing classification models that perform well on artificially balanced dataset suffer severe performance degradation on long-tailed datasets. This thesis presents three methods from different perspectives to address the issues in long-tailed visual recognition.

Key point sensitive (KPS) loss is proposed to address the model biased towards the head classes caused by long-tailed distributed data. KPS loss assigns relatively large margins on tail classes to relieve this bias. In addition, we find that key points are more important for classification. Therefore, KPS loss regularizes the key points strongly. Furthermore, the gradient signals of stimulus and inhibit samples for each class are re-balanced via the proposed gradient adjustment (GA) optimization strategy. This GA strategy can circumvent excessive negative signals on tail classes. KPS loss with GA significantly improves the overall classification accuracy on tail class with sacrificing a small amount of head class accuracy.

Feature-balanced loss (FBL) is proposed to address the limitation in KPS loss and study the effect of feature norm on classification. We observe that a large feature norm helps to achieve clear class margin and thereby proposing the novel FBL, which adds an extra class-based stimulus to the logit. The class based stimulus encourages large norm for tail classes. Moreover, the stimulus intensity is gradually increased in the way of curriculum learning. This robust training strategy helps to boost the classification performance and enable the model to be trained end-to-end. The proposed FBL incorporated with curriculum

learning achieve considerable performance gain on middle and tail classes meanwhile maintaining the competent performance in head classes.

This thesis further proposes the Gaussian clouded logit (GCL), which study the effect of softmax saturation on long-tailed learning. GCL perturbs different class logits with varied amplitudes to make the loss function with different degrees of softmax saturation for each class. The tail classes are set with relatively large amplitudes to decrease softmax saturation. Therefore, samples of tail classes are more active and their embedding space can be enlarged. To alleviate the bias in a classifier, the class-based effective number (CBEN) sampling strategy with classifier re-training is proposed, which can further improve the classification performance. GCL with CBEN achieves superior performance compared with the state-of-the-art methods.

Comprehensive evaluation and comparison are conducted on various benchmarks. Visualization experiments and discussion in depth of the proposed methods are provided. Experimental results demonstrate the superiority of the proposed methods.

Keywords: Long-tailed classification, Imbalanced learning, Loss modification, Logit adjustment

Acknowledgements

This thesis would not have been possible without the kind support and help of many individuals. I would like to express my sincere thanks to all those who have helped me.

First, I would like to give my heartfelt thanks to my supervisor, Prof. CHEUNG Yiu-ming, for his continuous support and kind guidance during my doctoral studies. Prof. CHEUNG not only guides us with a rigorous research attitude and strict academic regulations, but also demands that we should strive to do meaningful study and practical applications, rather than simply finishing papers. Besides, Prof. CHEUNG encourages us to maintain physical and mental health in life, which gives us the guarantee of continuous research. Prof. CHEUNG helped me adjust my research direction when I was confused, encouraged me when I was depressed, and gave me great support and help during Ph.D. study and research. His supervision and advice make my three and a half years Ph.D. experience more priceless and unforgettable.

I also appreciate having the opportunities to work with the colleagues in RRS735, DLB625 and ICTS for the stimulating discussions, sharing of study experiences and the happy time together. I would mention them in particular: Dr. LU Yang, Dr. YE Mang, Dr. PANG Meng, Dr. ZHANG Yiqun, Mr. ZOU Rong, Mr. YANG Lin, Mr. LIU Buhua, Mr. HU Zhikai, Mr. LI Ruiqi, Mr. JIANG Juyong, Mr. GONG Yikai, Miss LAN Weichao, Miss JI Hui, and Miss SUN Longxu.

I would like to thank to Hong Kong Baptist University, Department of Computer Science and Gradual School for providing us with such a pleasant environment to pursue

my research. A variety of interesting events and seminars organized by COMP enriched my academic experience.

I would also like to express my special thanks to my mother Ms. LI Wenchao and my family, who raised and supported me unconditionally. I also wish to thank my fiancé Mr. ZHAO Mingming and his family for understanding and encouraging me. Their love and company are always my everlasting power.

Last but not least, I feel really grateful to all those who spent a lot of time reading this thesis and gave me much constructive advice and comments, which will benefit me in my future study.

Table of Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

1.1 Background

Visual recognition problems have achieved considerable breakthroughs in a wide range of applications [1]–[5], owing that the deep Convolutional Neural Networks (CNNs) [1], [6], [7] are advanced and the large-scale, high-quality annotated datasets, for example, ImageNet ILSVRC 2012 [8] and Places database [9] are available. In such datasets, both the training and testing data have been artificially balanced. That is, each class has roughly the same amount of instances. However, from the practical point of view, the number of samples for different classes of data varies greatly due to the different difficulties in data collection. As a result, real-world datasets generally have skewed distributions with a long tail [10], [11], namely, a few dominant categories (called head classes) occupy most of the samples, while most of the remaining categories (called tail classes) are associated with rarely few samples, as Figure 1.1 shown. Nevertheless, a small sample size does not mean that the tail classes are unimportant. For example, when classifying mammals, training samples of endangered animals such as tigers and snow leopards are more difficult to obtain than those of common animals like cats and dogs. Nevertheless, endangered animals still need to be correctly classified given a query. Therefore, in order to show the equal importance of each class, their sample size should be roughly the same during the test stage even if the training data is long-tailed. Unfortunately, training on long-tailed data will raise a problem, *i.e.*, a biased learning process for the classification model, because

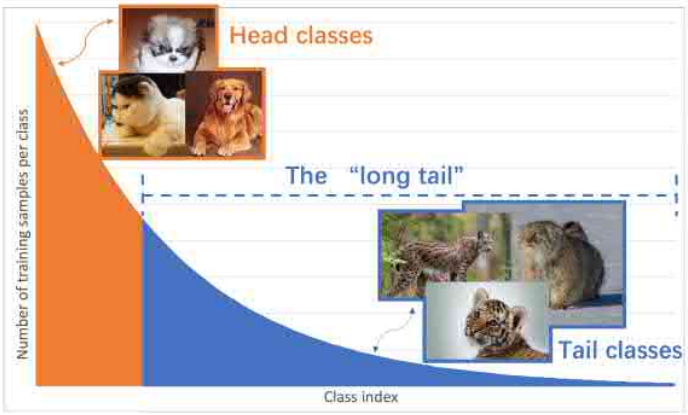


Figure 1.1: Real-world datasets generally have skewed distributions with a long tail. The severe imbalance poses a huge challenge for CNN models, particularly in the tail classes.

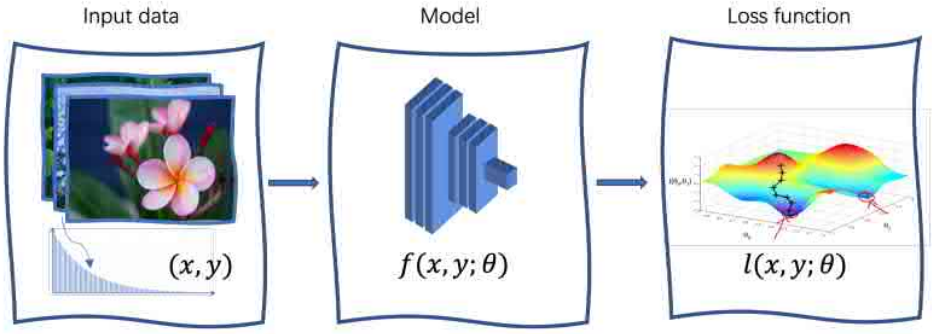


Figure 1.2: The existing approaches are grouped into three categories based on modifications in different stages of training.

the instance-rich head classes usually contribute to an overwhelmingly large quantity of negative samples for tail classes. Consequently, the learned classification model tends to have serious poor performance in the tail classes during testing.

1.2 Review of Methods for Long-tailed Classification

To solve the imbalance of long-tailed data, a mass of research has been conducted in recent years. As Figure 1.2 shown, the existing approaches can be divided into three categories based on the modifications in different stages of model training, *i.e.*, input data, model, and loss function. As the results, we categorize these methods as data refinement, module improvement and loss modification/logit adjustment.

1.2.1 Data Refinement

1.2.1.1 Data Re-balancing

One of the most straightforward methods is to re-sample the training data in a balanced manner [12]–[17]. Random over-sampling [13], [15], [18]–[20] the tail classes and random under-sampling the head classes are the two most common practices. However, in the case of severely imbalanced data distribution, random over-sampling which duplicates the head class samples will inevitably lead to overfitting [21]. Random under-sampling [22]–[24] that discards the head classes samples will improve tail classes accuracy by sacrificing that of head classes. Hybrid-sampling methods [25], [26] combining these two methods can overcome the shortcomings of the above two methods to a certain extent. Some other sampling strategies, for example, instance-balanced [12] sampling assigns equal probability (q_i) of being selected for each sample and class-balanced sampling [12] set q_i the same for each class i. Square-root sampling [27], [28] sets $q_i \propto n_i^{1/2}$, where n_i is the number of samples in class i. Progressively-balanced sampling [29] is a linear combination of instance-balanced and class-balanced sampling. Nevertheless, for large and severely imbalanced data, the effect of these samplers with a manually-defined sampling rate is limited due to the lack of tail class instances and the not necessarily optimal sampling rate for each class.

To address the aforementioned issues, recently, Wang *et al.* [30] re-designed the sampling rate through the dynamic curriculum learning to gradually increase the sampling rate of tail classes as training progresses. Ren *et al.* [31] proposed the Meta Sampler, which utilize a class-balanced set sampling from the original training set as the supervision to learn the optimal sampling rate for different classes. However, high sampling rate does not truly address the problem of severely sparse sample in tail classes. Zhang *et al.* [32] thereby developed the method called feature augmentation and sampling adaptation (FASA). It adjusts the sampling rate based on the class performance on validation set and meanwhile augments features virtually based on a distributional prior. FASA can effectively balance the data and avoid overfitting the tail classes. Liu *et al.* [33] self-paced harmonize [34] the

majority class data hardness via under-sampling. The the obtained subset is fed into an ensemble model to further improve the classification accuracy.

1.2.1.2 Re-weighting

Re-weighting methods assigns different importance for samples of different categories, and the importance will will be reflected in the weights of the loss function. The most intuitive way is to set the highest importance to the sample of the fewest number of classes, and thus the inverse of label frequency is directly utilized as the weight of the loss, *i.e.*, weighted softmax loss. In order to address the issue of overfitting the tail class samples with such loss, Cui *et al.* [35] introduced the novel "effective number" to replace label frequencies to re-weight the loss. Differently, Lin *et al.* [36] assign with large weights to the hard samples, namely, the ones that are most likely to be misclassified. Park *et al.* [37] proposed the influence-balanced loss, that fine-tunes the loss weights according the samples influence on class boundary instead of label frequencies. Besides these handcrafting weights, some researchers explored the learnable ways to seek the weights so that the training can be automatically and generally. For example, Ren *et al.* [38] proposed the L2RW that learns the weights of training instances based on their gradient. They found these weights through a meta gradient descent step in an on-line way. The parameters of the meta learner are obtained by minimizing the loss on balanced clean validation set. Shu *et al.* [39] proposed to formalize the weights on loss as an explicit weighting function. Then they utilizes a multi-layer perceptron to approximate the weighting function, the parameters of which are also updated based on a balanced validation set. The meta-learning paradigm does not require any additional hyperparameter tuning and thus is relatively easy to be implemented and can be transferable to other related tasks.

1.2.1.3 Data Augmentation

The goal of data augmentation is to use a number of strategies to increase the size and quality of training data for model to alleviate the under-representation for tail classes. The augmentation methods can be classified into input augmentation and feature augmentation.

Input augmentation increases sample diversity in the data space. The classical

augmentation methods [1], [40], [41] including flip, rotate, crop and padding *etc.*. Input mixup [42] is a simple but effective way that linearly combines the input images and their corresponding labels. Zhang *et al.* [43] and Zhong *et al.* [44] have proven that input mixup can deliver good representation in long-tailed learning. Following, ReMix [45] sets a disproportionately higher mixing factor for tail classes to force the classifier leave large generalization for minority class. Besides that, Wang *et al.* [46] use noise vectors to encode the variation information and further augment the minority classes. Differently, M2m [47] constructed a balanced dataset by translating head-class samples to the tail classes via another pre-trained classifier. Rare-class sample generator (RSG) [46] directly generates tail classes samples utilizing the variation information from frequent classes to augment tail classes.

Feature augmentation augments data diversity in feature space. Similar with the input mixup, there are several methods [43], [48]–[50] that linearly mixup the data in feature space. Knowledge transfer is another promising technology. Yin *et al.* [51] transferred the intra-class variance of head classes obtained from an encoder-decoder based network to augment the feature of tail class samples. Similar to it, Liu *et al.* [52] added the angular variance learned from the head class to the samples in tail classes to enlarge the intra-class diversity of tail classes. Chu *et al.* [53] utilize class activation maps (CAM) [54] to decompose the features into a class-generic and a class-specific component. Then, they enrich tail classes by fusing the class-specific components obtained from the tail classes with the class-generic components of the head classes. Zhang *et al.* [43] also exploited CAM. The use it to obtain the foreground in an image and then augment the obtained foreground object by flipping, rotating *etc.*. The augmented foreground is then simply placed on the unchanged background to obtain the new informative images.

1.2.2 Module Improvement

Decoupling representation [12] is a promising way. It obtains the representation in the first stage and then uses the data re-balancing or re-weighting method to re-train the classifier. The tail class performance has been significantly improved via the decoupling training

strategy. Similar this idea, Wang *et al.* [55] proposed a bi-level sampling scheme to calibrate the classifier. Based on decoupling representation, several methods have been proposed to improve the classifier. For example, Zhang *et al.* [56] proposed DisAlign to align the classifier to a class prior. Zhong *et al.* [44] proposed the MisLas to re-balance the classifier by label smoothing [57] which can also decrease over-confidence for classes. Another idea is to enhance the representation. Kang *et al.* [58] proposed the $mathcalk$-positive contrastive loss (KCL) that selects k samples from the same class to form a set of positive samples. KCL helps obtain more discriminative representations and thus can improve the classification accuracy.

Bilateral-branch network (BBN) [50] is another alternative method that can effectively enhance the model performance. It splits the original CNN into two branches, One of which focuses on the head classes by sampling the row long-tailed data, and the other branch focuses on tail data through reverse sampling. Then, BBN proposed a new cumulative learning strategy to make the network gradually pay attention to tail classes and manifold mixup is utilized to fuse the features obtained from these two branches. Finally, the classifiers from the two branches jointly decide the classification result. Following, Wang *et al.* [59] introduced contrastive learning [60], [61] to the bilateral-branch model which can further boost the performance on long-tailed data.

Ensemble learning systems intentionally build and merge numerous network modules. BBN [50] and its relevant methods (for example, see [55], [59], [62]) can also be seen as a kind of ensemble model that fuse the predicted results of two classifiers. Different with bilateral branches, balanced group softmax (BAGS) [63] and learning from multiple experts (LFME) [64] split the original long-tailed data into several subsets. Then, the subsets with similar class sizes are leveraged to train multiple experts, which can overcome classifier bias towards the frequent classes. Ally complementary experts (ACE) [65] divide the training data into several skill- diverse subsets instead of balanced ones. Test-time aggregating diverse experts (TADE) [66] cooperates with diverse re-sampling strategies to train the different experts which favor different kinds of distributions.

1.2.3 Loss Modification and Logit Adjustment

Re-margining adjusts the decision boundary to leave a large margin for tail classes via modifying the loss function. Cao *et al.* [67] proposed the Label-distribution-aware margin (LDAM) loss which introduces the class-based margin to the loss function to enlarge the tail class margins. They have also theoretically proved that the margin should be proportional to $-\frac{1}{4}$ power of the class size. Khan *et al.* [68] re-margin the loss function based on the class-level uncertainty obtained by Bayesian estimation. Wu *et al.* [69] proposed RoBal which exploits both class-specific bias and margins on the logits. It can also address the adversarial robustness under long-tailed distributed data.

The negative gradient over-suppression [70], [71] is one of the key issues for long-tailed data. Many researchers try to address the long-tailed problem from this perspective. Equalization loss [71] and equalization loss v2 [72] down-weight the negative gradients for model training through loss function and gradient, respectively. Droploss [73]) directly ignore the gradient from samples of head class for the tail class through the weights based on class sizes. Seesaw loss [74] introduces a mitigation factor and a compensation factor to the logit to re-balance positive and negative gradients for each class. The mitigation factor reduces the penalty for tail classes according to the proportion of the number of samples per class that the model received during training. In the mean time, the compensating factor increases the penalty applied to the relevant class when the false positive occurs. Adaptive class suppression loss (ACSL) [75] exploits the output confidence as an indicator to decide whether to suppress negative sample gradients. In details, if the predicted probability of a negative sample is less than a predefined threshold, the model should be confident, so this sample is useless and therefore its weight is set to 0 to avoid negative over-suppression. Otherwise, the model is still confused for the samples. The weight for the corresponding class should be reserved and therefore is set to 1.

Logit adjustment tunes the predicted logit based on different motivation. Menon *et al.* [76] mathematically showed that the margin should be Fisher consistent to minimize the balanced class error. Ren *et al.* [31] obtained the similar conclusion based on generalization

Table 1.1: Summary of Loss Function

Method	Loss function
Cross entropy	$L_{CE} = -\log p_y, p_y = \frac{e^{z_y}}{\sum e^{z_i}}$
Weighted softmax loss	$L_{WCE} = -\frac{1}{n_y} \log p_y, p_y = \frac{e^{z_y}}{\sum e^{z_i}}$
Focal loss (2017) [36]	$L_{focal} = -(1-p_y)^{\gamma_f} \log p_y, p_y = \frac{e^{z_y}}{\sum e^{z_i}}$
Effective number (2019) [35]	$L_{EN} = -\frac{1-\gamma_{EN}}{1-\gamma_{EN}^{n_y}} \log p_y, p_y = \frac{e^{z}_{y}}{\sum e^{z_i}}$
LDAM loss (2019) [67]	$L_{LDAM} = -\log p_y, p_y = \frac{e^{z_y - n_y^{-1/4}}}{e^{z_y - n_y^{-1/4}} + \sum_{i \neq y} e^{z_i}}$
Equalization loss (2020) [71]	$L_{Eq} = -\log p_y, p_y = \frac{e^{z_y}}{\sum w_i e^{z_i}}$
Balanced softmax (2020) [31] & Logit adjustment (2021) [76]	$L_{BS} = -\log p_y, p_y = \frac{e^{z_y - \log n_y}}{\sum e^{z_i - \log n_i}}$

Notations: z_i, logit of class i; and $i = y$ indicates target class; n_i, numbers of samples in class i; γ_*, hyper-parameter in method $*$.

error bound. When the model has seen more samples from head classes than tail classes, seesaw loss [74] has similar loss function with Menon [76] and Ren *et al.* [31]'s works. De-confound [77] exploited causal inference [78] to remove the "bad" causal effect in the logit.

Several kinds of loss functions and their formula representations are summarized in Table 1.1

The above three categories of methods are not strictly separated, they can be included, transformed or combined with each other. For example, decoupling representation based methods [12], [56] combines the balanced-sampling strategy. BBN [50] combines several different strategies, including re-sampling, manifold mixup [49] and curriculum learning [79]. LDAM [67] should be combined with a deferred re-balancing of re-sampling optimization schedule so that to effectively improve the tail class classification performance.

1.3 Major Contributions

To address the aforementioned long-tailed visual recognition problem, this thesis aims to exploit the technologies that do not increase model complexity and computational difficulty.

Therefore, how to modify the loss function and adjust the logits is our research focus. Based on this, we propose the following three works from different perspectives:

(1) **Key point sensitive (KPS) loss**: Chapter 3 addresses the model learning biased towards the head classes caused by long-tailed distributed data through regularizing the key points strongly and assigning relatively large margins on tail classes. In this way, the generalization performance of the classification model and the classification accuracy on tail classes can be improved. Furthermore, by virtue of the gradient analysis of the loss function, it is found that the tail classes always receive negative signals during training, which misleads the tail prediction to be biased towards the head. We therefore propose a gradient adjustment (GA) optimization strategy to re-balance the gradients of positive and negative samples for each class. The proposed GA strategy can circumvent excessive negative signals on tail classes and further improve the overall classification accuracy.

(2) **Feature-balanced loss (FBL)**: The feature-balanced loss introduced in Chapter 4 is proposed to address the limitation in KPS. KPS loss surrenders a small percentage of head class classification accuracy for considerably enhancing the performance of the middle and tail classes. FBL addresses the long-tailed problem from the perspective of feature norm. In details, it encourages larger feature norm of tail classes through adding an extra class-based stimulus to the logit. Moreover, the stimulus intensity is gradually increased in the way of curriculum learning. This robust training strategy not only helps to enhance the classification accuracy of tail classes to a large extent, but also maintains the performance of head classes. As a result, the FBL can be trained end-to-end and achieve considerable performance gain.

(3) **Gaussian clouded logit (GCL)**: Chapter 5 proposes GCL to better solve the long-tailed visual problem. The original cross-entropy loss can only propagate gradient short-lively because the gradient in softmax form rapidly approaches zero as the logit difference increases. This phenomenon is called softmax saturation, which is unfavorable for training on balanced data. GCL utilizes this seemingly harmful

phenomenon to balance the validity of the samples in long-tailed data, and thereby solving the distorted embedding space of long-tailed problems. Specifically, GCL Gaussian perturbs different class logits with varied amplitudes. The tail classes are set with relatively large amplitudes to make their samples more active as well as enlarge the embedding space. To alleviate the bias in a classifier, we accordingly propose the class-based effective number sampling strategy with classifier re-training, which can further improves the model performance.

1.4 List of Publications

The publications are listed as follow:

Journal papers

- **M. Li**, Y.-m. Cheung, and Z. Hu, "Key point sensitive loss for long-tailed visual recognition," *submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, under the second-round review.
- **M. Li** and Y.-m. Cheung, "Identity-preserved complete face recovering network for partial face image," *IEEE Transactions on Emerging Topics in Computational Intelligence (TETCI)*, pp. 1–6, 2021. DOI: 10.1109/TETCI.2021.3100646.
- M. Pang, Y.-m. Cheung, Q. Shi, and **M. Li**, "Iterative dynamic generic learning for face recognition from a contaminated single-sample per person," *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, vol. 32, no. 4, pp. 1560–1574, 2021. DOI: 10.1109/TNNLS.2020.2985099.

Conference papers

- **M. Li**, Y.-m. Cheung, and Y. Lu, "Long-tailed visual recognition via gaussian clouded logit adjustment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2022.
- **M. Li**, Y.-m. Cheung, and J. Jiang, "Feature-balanced loss for long-tailed visual recognition," in *Proceedings of the 2022 IEEE International Conference on Multimedia and Expo (ICME)*, Taipei, Taiwan, 2022.

- Y.-M. Cheung, **M. Li**, and R. Zou, "Facial structure guided gan for identity-preserved face image de-occlusion," in *Proceedings of the 2021 International Conference on Multimedia Retrieval (ICMR)*, ser. ICMR'21, Taipei, Taiwan, 2021, pp. 46–54. DOI: 10.1145/3460426.3463642.
- M. Pang, Y.-m. Cheung, Q. Shi, and **M. Li**, "Iterative dynamic generic learning for single sample face recognition with a contaminated gallery," in *Proceedings of the 2020 IEEE International Conference on Multimedia and Expo (ICME)*, 2020, pp. 1–6. DOI: 10.1109/ICME46284.2020.9102792.

1.5 Thesis Organizations

The rest of the thesis is organized as follows.

Chapter 2 introduces the basic notations used throughout this thesis, and provides the overview of five commonly used benchmark datasets utilized in the experiments.

Chapter 3 introduces the proposed KPS loss. We first review the feature points distribution in embedding space, and define three kinds of points based on their distance to anchor vectors, *i.e.*, key point, non-key point and simple point. Increasing the overall classification accuracy of the CNN model requires to make key points more separable. We therefore propose the KPS and give the motivation and mathematical analysis of it. Furthermore, we detail the propose GA optimization strategy which can can circumvent excessive negative signals on tail classes and further improve the overall classification accuracy. The extensive experiments are conducted on long-tailed benchmarks to compare with the state-of-the-art methods. The visualization experiments on the motivation of KPS and the classification results of KPS combined with other methods also presented.

Chapter 4 proposes a new loss modification method, namely, FBL, to address the limitation in KPS, namely, KPS loss surrenders a small percentage of head class classification accuracy for considerably enhancing the performance of the middle and tail classes. FBL address the long-tailed problem from the perspective of feature norm. In this chapter, FBL is detailed presented with its intuition, mathematical principles and algorithm. Experiments on long-tailed data are conducted to demonstrate the effectiveness of FBL and superior

classification accuracy in each classes.

Chapter 5 proposes the GCL to address the limitation in FBL, namely, the small performance gains compared to the latest proposed methods, and to better handle the long-tailed visual recognition problem. In this chapter, we observe that vanilla training on long-tailed data with cross-entropy loss makes the instance-rich head classes severely squeeze the spatial distribution of the tail classes, which leads to difficulty in classifying tail class samples. GCL provides a solution that utilizes softmax saturation to balance the sample effectiveness of each class. Then, the detailed mathematical derivation for the expression of the modified logit is given. Moreover, the corresponding re-balancing strategy, *i.e.*, CBEN is presented, followed by the proposed algorithm. We then present experimental results on widely used benchmark datasets to demonstrate the superior performance of GCL, which can outperform the most recent state-of-the-art counterparts by a notable margin.

Chapter 6 draws the conclusion of this thesis and outlines promising directions for future work.

The contents of Chapters 3, 4 and 5 have been partly published or submitted in the conferences/journals presented in Section 1.4 (by May 2022).

Chapter 2

Preliminaries

This chapter introduces the basic notations used throughout the thesis and the datasets used in the experiments.

2.1 Basic Notations

We define several common notations and terminology used throughout the thesis in this section. Further particular notations or definitions are supplied in each chapter accordingly.

2.1.1 For the dataset

In this thesis, let $\{x, y\} \in \mathcal{T}$ denotes a training sample from the training set $\mathcal{T}$, where $\mathcal{T}$ has totally C classes and N training samples, x represents the image that needs to be classified and $y \in \{1, \ldots, C\}$ is the ground truth label. For class j, $j = \{1, 2, \cdots, C\}$, the number of training samples is n_j and $\sum_{j=1}^{C} n_j = N$.

2.1.2 For the backbone

$\mathbf{f} \in \mathbb{R}^D$ denotes the representation of x obtained from the embedding layer and is the input of the last fully connected layer. D is the dimension of the feature. We use $\mathbf{z} \in \mathbb{R}^C$ to represent the output of the last fully connected layer of the CNN model. $\mathbf{W} = \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_c\} \in \mathbb{R}^{D \times C}$ represents the classifier weight matrix, namely, $\mathbf{z} = \mathbf{W}^T\mathbf{f}$. $\mathbf{z}$ is the vector composed of the logits of all classes. $\mathbf{w}_j$, $j \in \{1, \ldots, C\}$) represents the j-th column of $\mathbf{W}$ and is called class anchor vector, then $z_j = \mathbf{w}_j^T\mathbf{f}$ is the predicted logit of class j. We use the name *target logit* and *non-target logit* to represent $z_y = \mathbf{w}_y^T\mathbf{f}$ and

$z_j = \mathbf{w}_j^T \mathbf{f}, j \neq y$, respectively. The subscript y represents the ground truth class label here.

2.2 Datasets

Five commonly used benchmark datasets are with various scales, which include small-scale dataset: long-tailed CIFAR-10-LT and CIFAR-100-LT, and large-scale datasets: long-tailed ImageNet, iNaturalist and long-tailed Places.

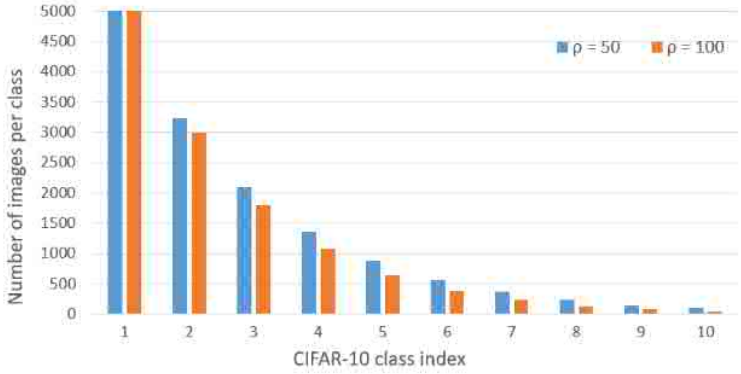
2.2.1 Long-tailed CIFAR

The original CIFAR-10 and CIFAR-100 datasets [80] are labeled subsets of the 80 million tiny images dataset, They were collected by Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton from a subset of 80 million tiny images dataset [81]. The CIFAR dataset can be divided into CIFAR-10 and CIFAR-100 according to the number of classified objects involved. This dataset is mainly used for image classification of deep learning and has been widely used. It consist of 50,000 color images of size 32×32 for training and 10,000 images with the same size for testing. CIFAR-10 and CIFAR-100 consists of 10 classes and 100 classes, respectively.

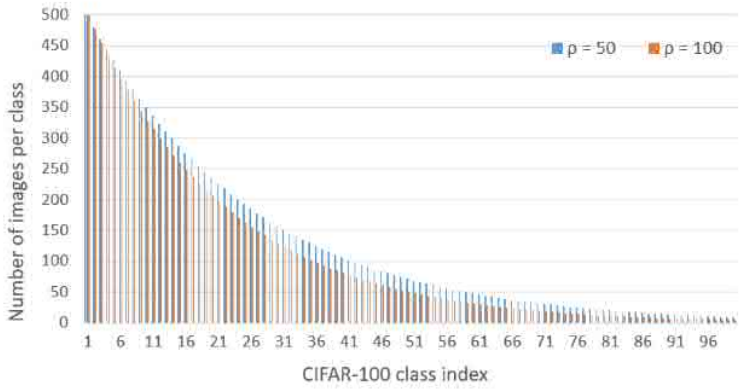
To create the long-tailed versions of CIFAR (CIFAR-10-LT and CIFAR-100-LT), we follow Cui *et al.*'s work [35] by down-sampling training images per class with the exponential function $n_i = n_{o_i} \times \mu^i$, where i is the class index (0-indexed), n_{o_i} is the number of training samples in original balanced CIFAR and $\mu \in (0, 1)$. The validation set remains unchanged. The imbalance ratio ρ is defined as $\rho = \frac{n_{max}}{n_{min}}$, where n_{max} and n_{min} are the sizes of the classes with the most and least training samples, respectively. In the literature, the most widely used ρ are 50, 100 and 200. Figure 2.1 shows number of training images per class on CIFAR-10-LT and CIFAR-100-LT with $\rho = 50$ and 100.

2.2.2 Long-tailed ImageNet

ImageNet dataset is a large-scale image dataset established to promote the development of computer image recognition technology. There were more than ten thousand images in the dataset, and each image was manually annotated with label (class name). The images



(a) Training data size per class for CIFAR-10-LT



(b) Training data size per class for CIFAR-100-LT

Figure 2.1: Amount of training data per class in manually created long-tailed CIFAR datasets with different imbalance ratios

in the ImageNet dataset cover most of the categories of images we can see in our life. The ImageNet dataset has been the benchmark for evaluating the performance of image classification algorithms. The 2012 version of ImageNet (ImageNet-2012)[8] is designed for classification, localization, *etc.*, which is one of the most commonly used versions. It contains more than 1.2 million images for training and 150K images for validation and testing.

We follow Liu *et al.*'s work [82] to construct the long-tailed version (ImageNet-LT) of ImageNet-2012 by truncating a subset with the Pareto distribution with the power value

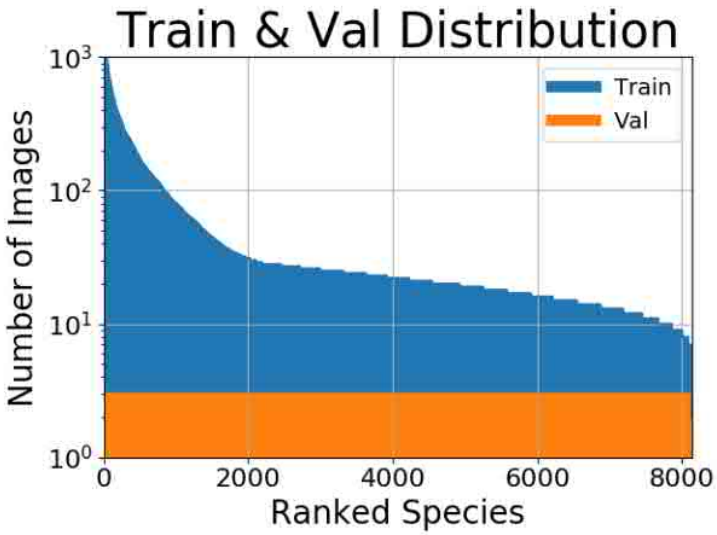


Figure 2.2: Training and validation distributions of iNaturalist 2018[1].

$\alpha = 6$ from the balanced version. Overall, this long-tailed version dataset has 1,000 categories with maximally 1,280 images and minimally 5 images per class. The original balanced validation data containing 50,000 images is used for validation in our experiments.

2.2.3 iNaturalist 2018

The iNaturalist Classification and Detection dataset (iNaturalist) [83] is composed of natural species. It consists of more than 8.5 million images from over 5 thousand variety species of plant, insects and animals. Each observation comprises of date, location, images, and labels identifying the species name in the corresponding images. The dataset includes visually similar species from all around the world. All the images were collected in a variety of conditions, including different camera types, varying image quality, *etc.*. The dataset features non-uniform distributions of images across item categories, and have been verified by multiple citizen scientists. iNaturalist is challenging to existing models due to its extremely imbalanced distribution and provides a good benchmark for the developments in computer vision.

[1]iNaturalist 2018 Competition: https://github.com/visipedia/inat_comp/tree/master/2018

Table 2.1: Overview of Long-Tailed Datasets

(a) On small-scale datasets

Dataset	CIFAR-10-LT			CIFAR-100-LT		
Number of Classes	10			100		
ρ	200	100	50	200	100	50
Number of Training image	11,203	12,406	13,996	9,502	10,847	12,608
Tail class size	25	50	100	2	5	10
Head class size	5,000	5,000	5,000	500	500	500
Number of validation image	-	-	-	-	-	-
Number of test image	10,000	10,000	10,000	10,000	10,000	10,000

(b) On large-scale datasets

Dataset	ImageNet-LT	Place-LT	iNaturalist 2018
Number of Classes	1,000	365	8,142
ρ	256	996	500
Number of Training image	115,846	62,500	437,513
Tail class size	5	5	2
Head class size	1,280	4,980	1,000
Number of validation image	20,000	7,300	24,426
Number of test image	50,000	36,500	-

iNaturalist has two versions, *i.e.*, iNaturalist 2017 and iNaturalist 2018. The 2017 version contains mostly species, but also had a few additional taxonomic ranks (*e.g.*, genus, subspecies, and variety). The 2018 version contains kingdom, phylum, class, order, family, and genus taxonomic information for all species. Our experiments adopt iNaturalist 2018, which contains more than 4.3 million training images from over 8 thousand categories. Figure 2.2 [84] shows training and validation distributions of it. We follow the official training and validation splits of iNaturalist 2018 in the experiments.

2.2.4 Long-tailed Places

The Places365 dataset [9] is designed for training artificial models which can be applied at a series high-level visual understanding tasks, for example, object recognition and classification, scene understand, event inference, to name a few. Places365 contains over 10 million images comprising more than 400 unique scene categories.

Long-tailed Places (Places-LT) is a long-tailed version of the Places365. There are

184.5K images with class sizes ranging from 5 to 4,980. Moreover, the gap between the sizes of tail and head classes of this dataset is larger than that of ImageNet-LT.

The datasets used in the experiments in the thesis and their features are summarized in Table 2.1.

Chapter 3

Key Point Sensitive (KPS) Loss for Long-tailed Visual Recognition

For long-tailed distributed data, existing classification models often learn overwhelmingly on the head classes while ignoring the tail classes, thus resulting in poor generalization capability. We therefore propose a new approach in this chapter to address the aforementioned problem, in which a key point sensitive (KPS) loss is presented to regularize the key points strongly to improve the generalization performance of the classification model. Meanwhile, in order to improve the performance on tail classes, the proposed KPS loss also assigns relatively large margins on tail classes. Furthermore, we propose a gradient adjustment (GA) optimization strategy to re-balance the stimulus and suppress gradients for each class. By virtue of the gradient analysis of the loss function, it is found that the tail classes always receive negative signals during training, which misleads the tail prediction to be biased towards the head. The proposed GA strategy can circumvent excessive negative signals on tail classes and further improve the overall classification accuracy. Extensive experiments conducted on long-tailed benchmarks show that the proposed method is capable of improving the classification accuracy of the model significantly in tail classes, while sacrificing a small amount of accuracy in head classes.

3.1 Introduction

To address the issue of imbalanced data distribution, a straightforward solution is re-balancing the data distribution of different classes to mitigate the extreme imbalance of the training set. In the literature, two of such representative techniques are re-sampling and re-weighting. Re-sampling methods usually sample the training images of different classes with different variants of sampling rates to make the class-wise sample sizes roughly balanced. Existing commonly used re-sampling methods include under-sampling ones (e.g., see [13], [18], [85], [86]), which randomly remove training samples from head classes, and over-sampling ones (e.g., see [19], [23], [87], [88]), which randomly replicate training samples from tail classes. Re-weighting methods (e.g., see [35], [89]–[91]) balance the contribution proportion of each classes to the classification model through the multiplicative parameters on the loss function. To increase the impact of tail classes in the training process, the multiplicative parameters are inversely proportional to the amounts of class samples. It is expected that these two techniques can make the distribution of training data closer to that of the testing data which is uniformly distributed. One limitation of the re-balancing methods, however, is the over-fitting on tail classes because of duplicated training on the tail class samples that provide essentially insufficient information. In the literature, there are several attempts (such as class-level re-weighting [72], [74], [75] and re-margining [67], [92], [93]) to alleviate this issue. For example, Cui *et al.* [35] proposed to re-weight the softmax cross-entropy loss by "effective number" of samples. Cao *et al.* [67] adopted both re-weighting and re-sampling techniques to train a CNN model. That is, they used a label-dependent regularizer to re-weight the tail classes stronger than the head classes and trained the network by a deferred re-sampling strategy.

Empirical studies have shown the success of the aforementioned approaches in their application domains, but they all ignore the differences between the points in embedding space and treat them equally. In fact, different kinds of points in embedding space may contribute differently to the classification model from the practical viewpoint. Hence, we define three kinds of points based on their importance. If the two classes that are most

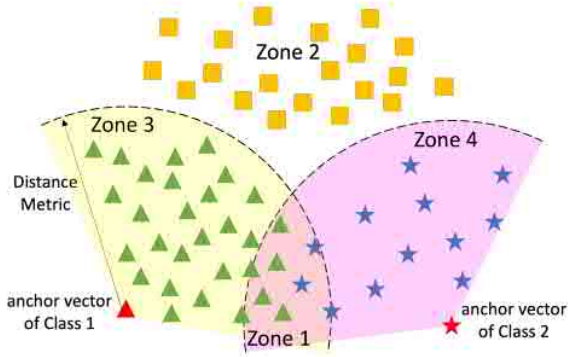


Figure 3.1: Different kinds of points in feature space. **Key points**: the points located in Zone 1 have small distance with the anchor vectors of both Class 1 and Class 2. **Non-key points**: the points falling in Zone 2 are far away from the anchor vectors of Class 1 and Class 2. **Simple point**: the points located in Zone 3 (or Zone 4) have smaller distance with the anchor vector of Class 1 (or Class 2).

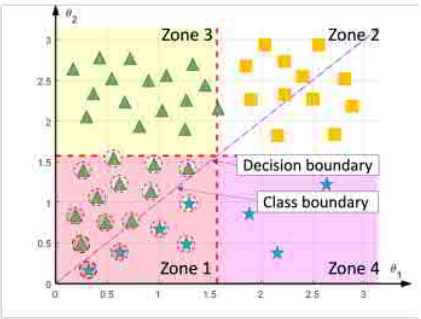
likely to be misclassified from each other can be predicted correctly, the remaining classes are relatively easy to be classified. Without loss of generality, we use these two classes for illustration. We name these two most difficult classes as Class 1 and Class 2, and the remaining classes are other classes. As shown in Figure 3.1, when assigning labels for the points in Class 1 and Class 2, the three kinds of points are

(1) **Key points**: The points located in Zone 1 have small distance with the anchor vectors of both Class 1 and Class 2. These points are most likely to be misclassified, because they are close to the anchor vectors of both Class 1 and Class 2. We call this kind of points as *key points*, which should be paid more attention.

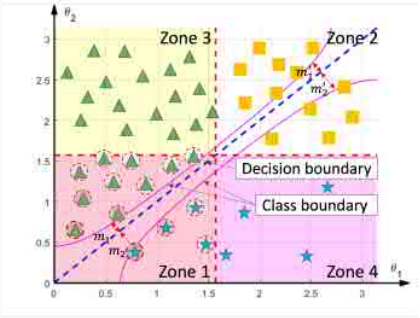
(2) **Non-key points**: The points falling in Zone 2 are far away from the anchor vectors of Class 1 and Class 2. This kind of points belong to other classes, because they are far away from anchor vectors of Class 1 and Class 2. We need not pay much attention to them. We call the points in Zone 2 as *non-key points*.

(3) **Simple points**: The points located in Zone 3 (or Zone 4) have smaller distance with the correct anchor vector. This kind of points is relatively easy to be classified. We name the points in Zone 3 and Zone 4 as *simple points*.

Increasing the overall classification accuracy of the CNN model requires to make key points



(a) Class boundary of softmax cross-entropy loss function.



(b) Class boundary of the loss function in Cao *et al.*'s work [67].

Figure 3.2: Class boundaries of different loss functions, where angular distance is used. The θ_1 axis represents angular distance between sample features and the anchor vector of Class 1, while the θ_2 axis is for the angular distance of Class 2. Shaded points with red textures represent key points. The denser the texture, the more important this is.

more separable. Therefore, key points should be regularized more strongly.

Based on the above analysis, we introduce the margins (*i.e.*the distance between the class boundary to the decision boundary[1]) to treat the aforementioned three kinds of points differently. In order to improve the separability of key points, their margin needs to be larger than that of non-key points. Unfortunately, most existing methods ignore this. Figure 3.2 shows the margins in the existing methods. For a clear visualisation, we use the angular distance as the distance metric and transform the points into the coordinate axis. We can see that most previous methods (for example, see [18], [45], [50]) use the basic softmax cross-entropy loss, which ignore margins. Their class boundary coincides with the decision boundary, which is shown in Fig. Figure 3.2a. In the literature, Cao *et al.*'s work [67] take the margins into consideration, but it allocates the same margins between key and non-key points, namely, $m_1 = m_1'$ and $m_2 = m_2'$ as shown in Figure 3.2b.

In this chapter, we therefore propose a key point sensitive (KPS) loss that makes the key points more separable as well as increases the model performance on tail classes. To make the samples with key points easier to separate, we regularize the key points strongly by

[1]Some papers, *e.g.*see [94], [95] and [96], name the boundary determined by training objective as decision boundary, while the decision boundary in [67] means the boundary defined by the standard cross-entropy function during testing. To make them distinguishable, we call the boundary determined by the training objective as class boundary and that defined by the standard cross-entropy function which is used during testing as decision boundary.

multiplying the proposed label-dependent factors on the predicted logits of the points. In order to increase the classification accuracy on tail class, the proposed loss also preserves a larger class margin between points of tail than head classes through adding a label distribution-based margin. Besides the modification of the loss function, we also design a gradient adjustment optimization strategy. In softmax cross-entropy form, those instances in head classes contribute a large quantity of negative samples to tail classes. In this case, the penalty signal overwhelmingly suppresses stimulus signal for tail classes, causing the bias in the learning process of the classifier. To balance the gradient signals, we propose the class distribution-based scale parameters to scaling down the overwhelming gradients of negative samples on the tail classes. Experiments on benchmark datasets have shown that the proposed KPS loss and optimization strategy can obtain higher classification accuracy for long-tailed data.

In summary, the main contributions of this chapter are highlighted as follows.

(1) We propose a key point sensitive loss, which can significantly improve tail class classification accuracy while compromising little on head classes. Considering that key points have a greater impact on overall accuracy than non-key points and a large margin helps improve the classification accuracy, KPS loss assigns large margins for key points and tail classes, which can increase the model classification accuracy on long-tailed datasets.

(2) We propose a gradient adjustment optimization strategy to prevent tail class from overwhelming suppression. In the softmax cross-entropy loss function, the penalty signal overwhelmingly suppresses stimulus signal for tail classes. Thus, we utilize class-based scale parameters to re-balances positive and negative gradient signals for each class, which can further improve the overall classification accuracy.

(3) We conduct extensive experimentation on several benchmarks. The results show that the new loss with the proposed gradient adjustment optimization strategy can significantly improve the performance of the model in tail classes, meanwhile maintaining the competent performance of the head classes.

3.2 Related Work

In this section, we make an overview of the most related works, which include deep metric learning methods, class re-balancing methods, and two-stage training methods. These methods all focus on obtain better feature representation.

3.2.1 Deep Metric Learning Methods

Deep Metric Learning aim to reinforce CNNs with more discriminative features. Intuitively, the learned features should maximize compactness within a class and separability between classes.

Liu *et al.* [97] proposed large-margin softmax loss, which adopted multiplicative margin on the the angles between sample features and class anchors. Chen *et al*. [96] proposed virtual softmax that injects a virtual negative class into the original softmax to enlarge the margin between classes. Wang *et al.* [5] proved that the feature norms of well-separated samples usually larger than others. Normalized features can help to eliminate the bias caused by bigger magnitudes of them. Thereby, they suggested measuring the similarity of two samples using the cosine distance of two feature. Liu *et al.*also proposed to use cosine similarity, but they only normalize the classifier weights and also add extra multiplicative angle margin to the modified softmax. After that, cosFace [98] and arcFace [95] are proposed, which utilize use additive margin in cosine space and angle space, respectively. These methods are experimentally demonstrated to achieve remarkable performance on balanced datasets, but suffer severe performance degradation in tail classes on when the training data is long-tailed.

3.2.2 Class Re-balancing Methods

The most commonly used class re-balancing methods are to balance the impact caused by class distribution differences, which include re-sampling, re-weighting and re-margining, to name a few.

Re-sampling methods include under-sampling the head classes [19], [23], [88]) and

over-sampling the tail classes [13], [18], [86], which are the most important two types. However, under-sampling that discards large amount of samples in head classes will deface the generalization ability when the imbalance ratio is extreme. Over-sampling which duplicates samples of tail classes usually causes over-fitting [50].

Re-weighting methods direct the network to allocate more attention to the samples in tail classes than head classes through the loss function, which usually assign large weights to tail classes[90]. Some methods like Focal loss [36], Meta-Weight-Net[39] and Cost-sensitive SVM [91], [99] can achieve fine-gained control though the sample dependent costs. Some methods (for example, see [90] and [100]) assign weights inversely proportional to the number of samples of different classes. However, re-weighting methods yield poor performance on head classes [35] and lead to optimization difficulty under the case of extremely imbalanced data and large-scale scenarios [101], [102]. Accordingly, Cui *et al.* [35] proposed to replace the sample frequency with the *effective number* of samples to re-weight the loss function. Recently, many works [71]–[75], [103] re-weight the loss function based on the gradients of different classes to overcome the negative gradient over-suppression. For example, to reduce the influence of negative samples for the tail classes, Tan *et al.* proposed equalised loss [71] and equalization Loss v2 [72], which introduced a weight term for each class on loss function and gradient, respectively.

Re-margining methods [67], [92], [93] adjust the decision margin towards different classes. For example, LDAM [67] increases the margin for tail classes, and decreases it for head classes based on class frequency. Feng *et al.* [92] increased the margin for rare class by the approximate mean classification logit.

The re-weighting and re-margining methods all have superior performance to the vanilla empirical risk minimization. Nevertheless, the loss functions of these methods do not consider the different influence of key and non-key points.

3.2.3 Two-stage Training Methods

Two-stage training strategy includes imbalanced training and balanced fine-tuning [43]. The first stage, namely the imbalanced training, utilizes the original long-tailed dataset to

train the network. The second stage usually uses re-sampling or re-weighting to fine-tune the network and should be applied with a small learning rate. Recently, Cao *et al.* [67] proposed LDAM to encourage large margin to the tail classes to improve the generalization. This method train the network with the original dataset at the first stage and apply DRW at the second stage. This LDAM with DRW training strategy significantly improved the performance on tail classes. Decoupling learning [12] and Bilateral-Branch Network (BBN) [50] both proposed to decouple the representation learning and classifier learning. Decoupling learning firstly uses imbalanced data for representation learning and then exploits balanced sampling strategy to adjust the classifier. BBN designs a dual-pathway structure and performs a fusion operation at the last layer of the network. One pathway of this model focuses on the head classes through directly sampling from the original imbalanced data, and the other pathway diverts the network attention to tail class through a reversed sampling of the data. Such decoupling of representation and classifier learning is another fruitful avenue of exploration.

In addition, many other methods of different learning paradigms, *e.g.* ensemble learning [64]–[66], [104], [105], meta-learning [82], [106], and knowledge transfer [101], [107], are also proposed to address long-tailed problems. These methods all show effectiveness, but significantly increase model parameters or optimization difficulty.

3.3 Preliminaries

Two lemmas used in this chapter are introduced as follows:

Lemma 3.1. *When the number of classes C is smaller than twice of the feature dimension D (namely $C < 2D$), any two class anchors can be distributed at least $\frac{\pi}{2}$ apart on a hypersphere of dimension D.*

Lemma 3.2. *For the binary classification case ($C = 2$), the loss function with margin of the target class is:*

$$\begin{aligned} \ell(1, \mathbf{f}) &= \frac{W_1}{T} \log(1 + e^{m_1} \cdot e^{-z}) \\ \ell(0, \mathbf{f}) &= \frac{W_0}{T} \log(1 + e^{m_0} \cdot e^{z}) \end{aligned}, \tag{3.1}$$

where $W_{\pm}, T > 0$ and $m_{\pm} \in \mathbb{R}$ are weights, temperature parameter, and margins. The loss in Equation (3.1) is Fisher consistent for the balanced error iff

$$\frac{W_1}{W_0} \cdot \frac{\sigma(T \cdot m_1)}{\sigma(T \cdot m_0)} = \frac{1-p}{p}, \tag{3.2}$$

where $\sigma(z) = (1 + \exp(-z))^{-1}$, $p = \mathbb{P}(y = 1)$ is the prior probability of $y = 1$.

Remark 3.1. The detailed proof of Lemma 3.1 and Lemma 3.2 can be found in [108] and [76], respectively.

3.4 Proposed Model: Key Points Sensitive Loss

3.4.1 Motivation

To simplify the interpretation, we use the two classes that are most likely to be confused by the classification model to elucidate the direct intuition of our KPS loss, because the remaining classes will be relatively easy to be classified when these two classes are separated. We name these two classes Class 1 and Class 2, respectively. Without loss of generality, we set Class 1 and Class 2 as head and tail class (*i.e.*$n_1 \gg n_2$), respectively. Considering we have a sample x_2 from class 2, correctly classifying x_2 requires:

$$z_2 > z_1. \tag{3.3}$$

Equation (3.3) can incorporate a margin to increase the class separability:

$$z_2 - m_2 > z_1, \tag{3.4}$$

where m_2 can be the label distribution dependent aware margin (LDAM) [67] that is chosen as $m_2 = \frac{C}{n_2^{1/4}}$ based on the empirical Rademacher complexity [109], C is a constant. Different with Equation (3.3), m_2 helps to assign relatively larger margin to tail class. From Equation (3.4), we find that a large predicted logit $z_i, (i = 1, 2)$ will weaken the function of the margin m_2. In order to maintain the consistency of the influence of margin on different

predicted logits, motivated by [94], we can normalize the output of the linear classifier:

$$\text{Norm}(z_i) = \frac{\mathbf{w}_i^T \mathbf{f}}{\|\mathbf{w}_i^T\|_2 \|\mathbf{f}\|_2} = \cos\theta_i, \tag{3.5}$$

where θ_i means the angle between class anchor vector W_i and feature f. Equation (3.5) means that cosine similarity is used to measure the distance between features and class anchor vectors. Then, Equation (3.4) can be changed to:

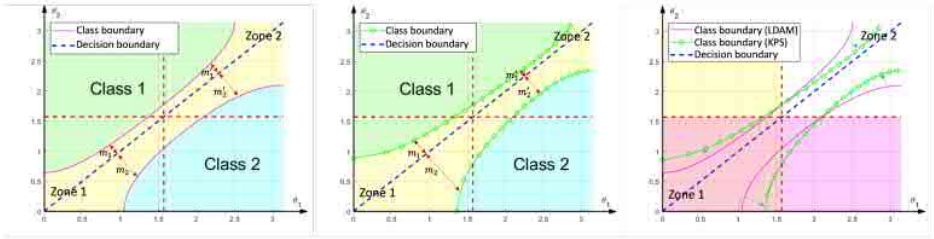
$$\cos\theta_2 - m_2 > \cos\theta_1. \tag{3.6}$$

Equation (3.6) decides the class boundary of LDAM for two classes, which can be seen in Figure 3.3a. It can be observed that the margins of different classes have two properties:

(1) The points fall in tail class (Class 2) has larger margins than head class, namely $m_2 > m_1$ and $m_2' > m_1'$;

(2) The points with different importance are both assigned the same margins, namely $m_1 = m_1'$ and $m_2 = m_2'$.

Intuitively, points with different importance should be treated differently. Because non-key point is with the angler distance larger than $\frac{\pi}{2}$ to both class anchor vectors (Zone 2 in Figure 3.3), it may not actually fall into either class as Lemma 3.1 indicated. Thus, there is no need to pay much attention to this kind of points. In contrast, the key points are close to both class anchor vectors (Zone 1 in Figure 3.3). If these points can be classified correctly, the classification accuracy can be improved. Therefore, we assign larger margins to the key points to make them easier to be classified. However, blindly increasing the margins results in compressing the feature space area of each class. In the extreme case, all samples in each class shrinks to one point, where the diversity of the features will disappear, leading to worse generalization. Take this into consideration, we set relatively small margin for non-key points, namely, $m_1 > m_1'$ and $m_2 > m_2'$. Non-key points are too far away from both classes anchor vectors to be assigned to each class according to Lemma 3.1. Therefore, assigning small margins of these points does not affect the overall classification accuracy. We can move the class boundary as Figure 3.3b shown without



(a) LDAM (b) KPS (c) Comparison

Figure 3.3: Margins of different loss functions and their comparison under binary-classes scenarios, where the θ_1 and θ_2 axes represent the angular distance between the sample features and the class anchor vectors of $\mathbf{w}_1$ and $\mathbf{w}_2$, respectively, and $m_{1,2}$ and $m'_{1,2}$ represent class margins.

compressing the feature space area and reducing feature diversity. Figure 3.3c shows the comparison before and after the movement of class boundary.

3.4.2 KPS Loss

3.4.2.1 KPS Loss for Two Classes

We can observe that the margins of Equations (3.3), (3.4) and (3.6) are the same for all points in the logit space because their slopes of the class boundary in their corresponding logit space are equal to 1. We prefer to assign a larger margin for key points, *i.e.*those points that are close to the anchor vectors of both classes. That is, points with larger logits have larger margins, thus we introduce a label-dependent factor r_i called radius to make the slope of the class boundary non-1. As a result, m_i $(i = 1, 2)$ shown in Figure 3.3 can be enlarged and the key points can be more separable. Accordingly, the class boundary function is rewritten as:

$$r_2 \cdot \cos\theta_2 - m_2 > r_1 \cdot \cos\theta_1.$$

The class boundary defined by Section 3.4.2.1 is shown in Figure 3.3b. Suppose $\psi(n)$ represents a non-decreasing function. The margin $m_i \propto 1/\psi(n_i)$ can encourage relative large margins for tail classes. The property 1, namely, $m_2 > m_1$ and $m'_2 > m'_1$ can be maintained to ensure the good performance of the model in the tail class. We expect that the features of tail class samples can be clustered tightly around the class anchor vector

so that they can be correctly classified relatively easily. Therefore, $\cos\theta_i$ for tail classes should be relatively large. To encourage large $\cos\theta_i$ for tail classes and regularize key points strongly at the same time, we set $r_i \propto \psi(n_i)$. The class boundary leaves larger margin for key points, namely, $m_1 > m_1'$ and $m_2 > m_2'$.

3.4.2.2 Selection of Margin and Radius

According to Cao *et al.* [67], the margin can be chosen as $m_i = \frac{C}{n_i^{1/4}}$ for class i. However, Menon *et al.* [76] pointed out that this margin is not fisher consistent for minimising the balanced error. We consider to obtain the expression of m_i based on Lemma 3.2. Then, the following equation can be obtained:

$$\begin{aligned} \frac{1+e^{-m_1}}{1+e^{-m_2}} &= \frac{1-p}{p} \Leftrightarrow \\ \frac{1+e^{-m_1}}{1+e^{-m_2}} &= \frac{n_1/N}{n_2/N} \Leftrightarrow \\ \frac{1+e^{-m_1}}{1+e^{-m_2}} &= \frac{n_1}{n_2}, \end{aligned} \tag{3.7}$$

where $p = \mathbb{P}(y=2)$. Suppose $C_t \cdot n_1 \gg 1$, we can have:

$$\begin{aligned} m_1 &= -\log(C_t \cdot n_1 - 1) \\ &\approx -\log(C_t \cdot n_1). \end{aligned} \tag{3.8}$$

Since $m_{1,2} > 0$, we add a constant $C' > 0$ to Equation (3.8):

$$\begin{aligned} m_1 &= -\log(C_t \cdot n_1) + C' \\ &= C_m - \log n_1, \end{aligned} \tag{3.9}$$

where C_m should satisfy $C_m \in \mathbb{R}$ and $C_m \geq \log n_{max}$, n_{max} is the largest class size. Eventually, we choose the general expression of margin m_i as:

$$m_i = C_m - \log n_i. \tag{3.10}$$

Since radius r_i has opposite monotonicity with m_i, for simplicity, we set:

$$r_i = \log n_i + C_r, \tag{3.11}$$

where C_r is a constant.

3.4.2.3 Extending KPS Loss to Multiple Classes

We use $\tilde{z}_j$ to represent the modified score of the j-th class in Section 3.4.2.1 for convenience, namely,

$$\tilde{z}_j = r_j \cos\theta_j. \tag{3.12}$$

To extent the loss to multi-class case, we first extent the binary classification loss to hinge loss:

$$L_{KPS-hinge}(x, y) = \max(\max_{j \neq y}\{\tilde{z}_j\} - \tilde{z}_y + m_y, 0). \tag{3.13}$$

This loss bring only one positive gradient and one negative gradient enter the network each time during optimization, which makes the convergence speed very slow when C is large. We can use LogSumExp function to replace the $\max$ function:

$$L_{KPS-lse}(x, y) = \max\left(\log\left(\sum_{j=1, i\neq y}^{C} e^{\tilde{z}_j}\right) - \tilde{z}_y + m_y, 0\right). \tag{3.14}$$

Notice that softplus function $\log(1 + e^x)$ can smoothly relax $\max(x, 0)$ and further improve generalization of the loss function. We can substitute $\max(x, 0)$ with softplus function and then obtain the following expression of KFS loss for multiple classes:

$$\begin{aligned} L_{KPS}(x, y) &= \log(1 + e^{\log\left(\sum_{j=1, i\neq y}^{C} e^{\tilde{z}_j}\right) - \tilde{z}_y + m_y}) \\ &= \log(1 + \frac{\sum_{j=1, i\neq y}^{C} e^{\tilde{z}_j}}{e^{\tilde{z}_y - m_y}}) \\ &= -\log\frac{e^{\tilde{z}_y - m_y}}{e^{\tilde{z}_y - m_y} + \sum_{j=1, j\neq y}^{C} e^{\tilde{z}_j}} \end{aligned}. \tag{3.15}$$

This is in the form of the well-known cross-entropy loss. However, the modified logit $\tilde{z}_j$ is small, which is not conducive to convergence. As suggested in [5] and [110], we use a large number s to scale $\tilde{z}_j$. Eventually, the loss function is expressed as:

$$L_{KPS}(x, y) = -\log\frac{e^{s\cdot(r_y \cos\theta_y - m_y)}}{e^{s\cdot(r_y \cos\theta_y - m_y)} + \sum_{j=1, j\neq y}^{C} e^{s\cdot r_j \cos\theta_j}}. \tag{3.16}$$

3.4.2.4 Analysis of KPS Loss and Comparison with Previous Methods

Bring the radius (Equation (3.11)) and the modified logit (Equation (3.12)) back to KPS loss for multiple classes (Equation (3.16)), because

$$\begin{aligned} e^{r_i \cdot \cos\theta_i} &= e^{\log n_i \cdot \cos\theta_i + C_r \cos\theta_i} \\ &= {n_i}^{\cos\theta_i} \cdot e^{C_r \cos\theta_i} \\ &= \left(n_i \cdot e^{C_r}\right)^{\cos\theta_i} \end{aligned} . \tag{3.17}$$

Therefore, we have

$$\begin{aligned} L_{KPS}(x,y) &= -\log \frac{e^{r_y \cos\theta_y - m_y}}{e^{r_y \cos\theta_y - m_y} + \sum_{j=1, j\neq y}^{C} e^{r_j \cos\theta_j}} \\ &= -\log \frac{\left(n_y \cdot e^{C_r}\right)^{\cos\theta_y} \cdot e^{-m_y}}{\left(n_y \cdot e^{C_r}\right)^{\cos\theta_y} \cdot e^{-m_y} + \sum_{j=1, j\neq y}^{C} \left(n_j \cdot e^{C_r}\right)^{\cos\theta_j}} \\ &\propto -\log \frac{\left(n_y \cdot e\right)^{\cos\theta_y} \cdot e^{-m_y}}{\left(n_y \cdot e\right)^{\cos\theta_y} \cdot e^{-m_y} + \sum_{j=1, j\neq y}^{C} \left(n_j \cdot e\right)^{\cos\theta_j}} \end{aligned} . \tag{3.18}$$

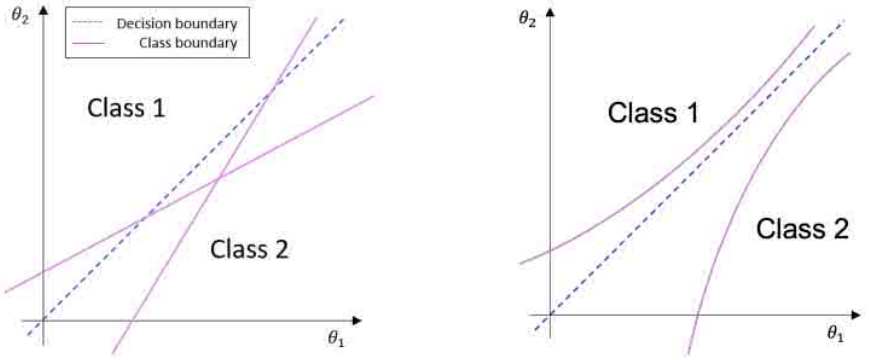
We can notice that KPS loss re-weights the loss through the output probability of different classes. In Equation (3.18), the item $n_j \cdot e$ acts as an attractor that requires the tail classes to be clustered closer than the head classes. The item e^{-m_y} leaves larger margin for tail classes. It is intuitively more reasonable because the hidden space capacity is finite.

In comparison with the previous methods, the proposed KPS loss has three characteristics that make it more appropriate for long-tail datasets:

(1) KPS loss adopts class-based margin which assigns relatively large margin for tail class. Previous methods [95], [98], [111] do not consider severe class imbalance data and set the same margin for all classes.

(2) KPS loss adopts cosine similarity to avoid bias caused by feature norm and class anchor norm. The margin modification losses [31], [74] utilize the inner product to measure the similarity between the feature and class anchor, but a feature or an anchor vector with large norm will cause privilege.

(3) KPS loss makes the slopes of different class boundaries different in the logit space, so that larger margins can be assigned to the points with higher logits (*i.e.*, the



(a) angle penalty (b) cosine penalty

Figure 3.4: Schematic plot of angle penalty and cosine penalty.

key points). In this way, samples that are prone to misclassification can be further separated. The prior margin modification methods [31], [67], [74], [76] utilize additive margin. The slopes of the different class boundaries are all equal to 1. The margins are the same for different kinds of points.

In addition, inspired by [95], the margin and radius can also be applied on the angle. Let us take the binary classification case as an example. Accordingly, the class boundary is then expressed as:

$$\hat{r}_2\theta_2 + \hat{m}_2 < \hat{r}_1\theta_1. \tag{3.19}$$

We can combine the angle penalty and cosine penalty in a unified framework with r_i, $\hat{r}_i$, m_i and $\hat{m}_i$, $(i = 1, 2)$ as the hyper-parameters. Hence, we have:

$$r_2 \cdot \cos(\hat{r}_2\theta_2 + \hat{m}_2) - m_2 > r_1 \cdot \cos(\hat{r}_1\theta_1). \tag{3.20}$$

The class boundary of angle penalty and cosine penalty are shown in Figure 3.4. For cosine penalty, we can use the margin m_i to avoid the intersection of class boundary and decision boundary, but class boundary of angle penalty always has intersect with decision boundary. In addition, the cosine penalty is easier to calculate than the angle penalty, because the acquisition of the angle θ_i needs to calculate the arc cosine. Thus we choose the cosine penalties, *i.e.*Section 3.4.2.1), as the class boundary which is equivalent to $\hat{r}_i = 1$ for $i = 1, 2, \hat{m}_2 = 0$ in Equation (3.20).

3.4.3 Gradient Adjustment Optimization Strategy

3.4.3.1 Analysis of Gradient

For a given sample x with the label y, we use $\hat{z}_i$ to represent the final logit of sample x belonging to class i, namely,

$$\hat{z}_i = \begin{cases} r_i \cos\theta_i - m_i, & i = y \\ r_i \cos\theta_i, & i \neq y \end{cases}. \tag{3.21}$$

The loss function is:

$$L_{KPS}(x, y) = -\log(p_y), \text{ with } p_y = \frac{e^{s \cdot \hat{z}_y}}{\sum_{j=1}^{C} e^{s \cdot \hat{z}_j}}. \tag{3.22}$$

For a given training sample, the gradients on $\hat{z}_i$ is:

$$\frac{\partial L_{KPS}}{\partial \hat{z}_i} = \begin{cases} s \cdot (p_i - 1), & i = y \\ s \cdot p_i, & i \neq y \end{cases}. \tag{3.23}$$

Equation (3.22) demonstrates that samples of class y punish the classifier of other classes *w.r.t.* $s \cdot p_i, (i \neq y)$. If class y belongs to the head class, it will contains enormously greater instance number than that of tail classes. Then, the classifier of tail classes will receive penalties in most samples and rarely get positive signals. In this case, the predicted probabilities of tail classes are severely suppressed, leading to a low classification accuracy.

To alleviate this gradient over-suppression problem, some previous works (*e.g.*, see [71], [73], [75]) directly ignore the gradient from samples of head class for the tail class by the weights based on class sizes. The predicted probability $\tilde{p}_i$ of class i given by the loss function of these methods is:

$$\tilde{p}_i = \frac{e^{z_i}}{\sum_{j=1}^{C} \tilde{w}_j e^{z_j}}, \tag{3.24}$$

where $\tilde{w}_j, j \neq i$ is 1 for head class and less than 1 for tail class. Equation (3.24) can achieve:

(1) For tail classes, only positive samples propagate gradients;

(2) For head classes, the negative gradients propagated to tail classes are reduced.

However, the sum of predicted probability for all classes of one training sample is not equal to 1, *i.e.*$\sum_{i=1}^{C} \tilde{p}_i \neq 1$. It does not conform to the property of probability.

In this chapter, we also introduce an adjustable scale parameter s_i for class i to re-balance the gradient signal of each class. Because we want to decrease the gradients of head classes and increase that of the tail classes in Equation (3.23) during optimization, s_i should be set to a decreasing number with respect to n_i, which has the same trend with m_i. Accordingly, we set:

$$s_i = C_s \cdot m_i, \tag{3.25}$$

for the constant $C_s > 0$. The KPS loss with gradient adjustment (GA) optimization strategy is:

$$L_{KPS-GA}(x, y) = -\log(\hat{p}_y), \text{ with } \hat{p}_y = \frac{e^{s_y \cdot \hat{z}_y}}{\sum_{j=1}^{C} e^{s_y \cdot \hat{z}_j}}. \tag{3.26}$$

The predicted probability given by Equation (3.26) satisfies that $\sum_{i=1}^{C} \hat{p}_i = 1$.

3.4.3.2 Analysis of Scale Parameters

For the original $p_i = \frac{e^{s \cdot \hat{z}_i}}{\sum_{j=1}^{C} e^{s \cdot \hat{z}_j}}$, it is well known that the smaller the scale parameter s is, the more uniformly p_i is distributed. As s increases, p_i will quickly decay for non-maximum logits and approach 1 for the maximum logit. The extreme cases are:

(1) When $s = 0$, $p_i = \frac{1}{C}$ for $i = \{1, 2, \cdots, C\}$;

(2) When $s \to \infty$, $p_i = \begin{cases} 1, & i = M \\ 0, & i \neq M \end{cases}$, where M is the class with the maximum logit.

If s is relatively small, there will be more classes that provide gradients, which means more non-target classes will vote for the class boundary. When s is large, only classes with high logits will provide gradients. That is to say, the samples that prone to be misclassified will participate in class boundary voting. In the early stage of training, we hope that all samples participate in the training to the same degree in order to converge quickly. Besides, the risk of tail class false positives will be increased if we simply shrink the gradient of head class samples, since samples from other classes that are misclassified as tail classes are penalized less. Therefore, we set s_i at the same value for all classes in the early stage of training. When the model converges to a certain extent, we calculate s_i by Equation (3.25)

Algorithm 3.1: KPS loss with Gradient Adjustment Optimization.

Input: Training dataset $\mathcal{T}$;
A CNN network $\phi((x, y); \omega)$ which is parameterized by ω;
Output: Predicted labels;

for $iter = 1$ *to* I_0 **do**
  Sample batch samples $\mathcal{B}$ from $\mathcal{T}$ with batch size of b;
  Calculate the loss by Equation (3.15): $\mathcal{L}((x, y); \omega) = \frac{1}{b} \sum_{(x,y)\in\mathcal{B}} L_{KPS}(x, y)$;
  $\omega = \omega - \alpha \nabla_\omega \mathcal{L}((x, y); \omega)$;
end
for $iter = I_0 + 1$ *to* I_1 **do**
  Sample batch samples $\mathcal{B}$ from $\mathcal{T}$ with batch size of b;
  Calculate the loss by Equation (3.26):
  $\mathcal{L}((x, y); \omega) = \frac{1}{b} \sum_{(x,y)\in\mathcal{B}} L_{KPS-GA}(x, y)$;
  $\omega = \omega - \alpha \nabla_\omega \mathcal{L}((x, y); \omega)$;
end

for class i. The network will give more positive signals to the tail class in this way. At the same time, the classifier can focus more on distinguishing the samples from tail classes that are prone to be confused.

The overall training procedure is summarized in Algorithm 3.1.

3.4.4 Time-complexity Analysis

For a given sample, the time-complexity of the original softmax cross-entropy loss is $\mathcal{O}(C)$ (C is the total number of classes), namely it is linear with the input dimension of the loss function. Equation (3.15) and Equation (3.26) show that KPS loss only adds scalar addition and multiplication compared with the original softmax cross-entropy loss. Therefore, KPS loss has $\mathcal{O}(C)$ time-complexity, which only adds negligible burden on the training process.

3.5 Experiments

3.5.1 Basic Setting

We use Pytorch [112] to implement and train all backbones from scratch by stochastic gradient descent (SGD) with momentum of 0.9.

3.5.1.1 Basic Setting on CIFAR-10/100-LT

For CIFAR-10-LT and CIFAR-100-LT datasets, the same data augmentation strategies with [1] is utilized, namely, randomly cropping a 32×32 region from the image that is flipped with 0.5 probability and padded with 4 pixels on each side. ResNet-32 is chosen as the backbone network with weight decay of 2×10^{-4}. The total number of training epochs is 200 with the mini-batch size of 64. Learning rate is initialized to 0.1 and multiplied by 0.01 at the 160-th and 180-th epoch, respectively. The linear warm-up learning rate [41] is used in the first five epochs.

3.5.1.2 Basic Setting on ImageNet-LT and iNaturalist 2018

For ImageNet-LT and iNaturalist 2018, we follow the data augmentation strategies in [41], namely, firstly scaling the shorter dimension to 256 and randomly cropping a 224×224 patch from the augmented image or its horizontal flip. For fair comparisons, we adopt the most common practices which are ResNet-50 [1] and ResNeXt-50 [7] for ImageNet-LT, and ResNet-50 for iNaturalist 2018, respectively.

The weight decay in all experiments on ImageNet-LT and iNaturalist 2018 is set as 1×10^{-4}. We train the network for 180 epochs. During training, we decay the learning rate by 0.1 at the 120-th and 160-th epoch, respectively. The mini-batch size is set as 128 and the learning rate is initialized to 0.1 for ImageNet-LT. As for iNaturalist 2018 dataset, we use the mini-batch size of 512 to speed up training. The learning rate should be increased by the square root of the mini-batch size according to [113], so the learning rate is initialized to 0.2.

The basic setting is summarized in Table 3.1.

3.5.2 Hyper-parameters selection

There are hyper-parameters in r_i, m_i and s_i. In order to simplify the selection of hyper-parameters for different datasets, we adopt the following strategy.

For r_j, we firstly normalize the minimum number in the class number list $\mathbf{n} =$

Table 3.1: Summary of Basic Setting for KPS

Dataset	CIFAR-10-LT	CIFAR-100-LT	ImageNet-LT	iNaturalist 2018
Backbone	ResNet-32	ResNet-32	ResNet-50, ResNeXt-50	ResNet-50
Mini-batch size	64	64	128	512
Initial lr	0.1	0.1	0.1	0.2
Weight decay	2×10^{-4}	2×10^{-4}	1×10^{-4}	1×10^{-4}
lr warm-up	Yes	Yes	No	No
Maximum epochs	200	200	180	180
lr decay ratio	0.01	0.01	0.1	0.1
lr decay epochs	{160,180}	{160,180}	{120,160}	{120,160}

$\{n_1, n_2, \cdots, n_C\}$ to a preset number n_{base} to get a temp list $\mathbf{r}' = \{r'_1, r'_2, \cdots, r'_C\}$:

$$r'_i = \log\left(n_i \times \frac{n_{base}}{n_{min}}\right),$$

where, n_{min} represents the minimum number of $\mathbf{n}$. Then, we obtain the final list $\mathbf{r} = \{r_1, r_2, \cdots, r_C\}$ by normalizing the minimum number in $\mathbf{r}$ to 1:

$$r_i = r'_i \times \frac{1}{r'_{min}}, \tag{3.27}$$

where r'_{min} is the minimum numbier in the temp list $\mathbf{r}'$.

For m_i, we first get a temp list $\mathbf{m}' = \{m'_1, m'_2, \cdots, m'_C\}$ based on Equation (3.10) by:

$$\mathbf{m}'_i = r_{max} + 1 - r_i, \tag{3.28}$$

where r_{max} is the maximum number in $\mathbf{r}$.

Then, the final margin list $\mathbf{m} = \{m_1, m_2, \cdots, m_C\}$ is obtained by normalizing the maximum number m'_{max} in $\mathbf{m}'$ to m_{max}:

$$m_i = m'_i \times \frac{m_{max}}{m'_{max}}. \tag{3.29}$$

For s_i, we obtain the scale parameter list $\mathbf{s} = \{s_1, s_2, \cdots, s_C\}$ by:

$$\mathbf{s} = s_{base} \cdot \mathbf{m}', \tag{3.30}$$

where s_{base} is the scale parameter in the first stage, which is same for all classes.

In this way, The parameters that need to be preset are: n_{base}, m_{max} and s_{base}. We choose

$n_{base} = 50$ and $s_{base} = 15$ experimentally for all datasets. The only hyper-parameter that need to be adjusted towards different dataset is m_{max}. For CIFAR-10/100-LT datasets, the total classes C are much smaller than the dimension of features. So the margin for CIFAR-10/100-LT can be set relatively large. We choose $m_{max} = 0.5$ for these two datasets. As for the large-scale datasets, namely, ImageNet-LT and iNaturalist 2018, we select $m_{max} = 0.35$ and 0.3, respectively.

3.5.3 Comparison Methods

We compare our KPS loss to the following three groups of competing methods that are most related to our method:

- **Baseline methods**. Besides vanilla training with cross-entropy loss (CE loss), we also employ the recent proposed focal loss [36] that focuses training on difficult samples as one of our baselines.
- **Loss modification methods**. Four recently proposed loss function modification methods are compared: CosFace [98], ArcFace [95], Label-Distribution-Aware Margin Loss with deferred re-weighting (LDAM-DRW)[3] [67], Class-Balanced focal loss (CB-Focal) [35], Equalised loss [71] and Logit Adjustment loss (LA loss) [76]
- **Two-stage methods**. We compare with the most recently proposed Bilateral-Branch Network (BBN) [50] and meta-learning [106], which adopt two-stage fine-tuning strategy and a domain adaptation strategy, respectively. These two methods achieve good performance on those four aforementioned commonly used long-tailed datasets. For ImageNet-LT and iNaturalist 2018 datasets, we also compare with Decoupling learning [12], which takes the strategy with post-hoc normalisation of the classification weights and achieves high accuracy on large-scale datasets.

3.5.4 Comparison Results

Top-1 error rates of baseline and comparison methods are shown in Table 3.2 and Table 3.3. We use the results reported in references ([35], [36], [50], [67], [71], [76], [106]) except

[3]LDAM-DRW trains the backbone with the original dataset at the first stage and fine-tune the classifier with DRW at the second stage. It also belongs to two-stage method.

Table 3.2: Comparison on Long-tailed CIFAR Datasets.

Dataset	CIFAR-10-LT		CIFAR-100-LT	
Backbone	ResNet-32			
Imbalance ratio	100	50	100	50
CE loss	29.64	24.78	63.32	56.15
Focal loss [36]	29.62	24.75	62.75	55.68
CosFace [98]	27.92	22.60	60.79	56.89
ArcFace [95]	26.24	21.81	60.94	56.60
LDAM-DRW [67]	22.97	18.97	57.96	53.41
CB-Focal [35]	25.43	20.73	60.40	53.79
Equalised loss*[71]	26.02	-	57.26	-
LA loss [76]	**19.08**	-	56.11	-
BBN [50]	20.18	17.82	57.44	52.98
Meta-learning*[106]	20.00	17.77	55.92	50.84
Meta-learning†[106]	21.10	**17.12**	**55.30**	**49.92**
KPS loss (Ours)	**18.77**	**15.41**	**54.97**	**50.82**

Note: Top-1 error rates (%) are presented. The best and the second best results are shown in **underline bold** and **bold**, respectively.
* Results are obtained by coping from LA loss [76].
* Results are obtained by incorporating LDAM [67].
† Results are obtained by incorporating focal loss [36].

the baseline method. Our re-implemented results are mostly consistent with references. The slightly inconsistent ones might be caused by running environment (e.g., the version of CUDA and precision of device), since we keep the experimental settings consistent with the references.

3.5.4.1 Experimental Results on CIFAR-10/100-LT

We conduct extensive experiments on CIFAR-10-LT and CIFAR-100-LT with different imbalance ratios. The performance comparison of various methods is shown in Table 3.2. Our proposed KPS loss achieves the best results in most cases compared other methods, including loss modification strategies (*i.e.*LDAM-DRW, CB-Focal, Equalised loss and LA loss), two-stage fine-tuning strategy (*i.e.*BBN), and also two-stage domain adaptation strategy (*i.e.*meta-learning). These methods are all recently proposed state-of-the-arts.

Except meta-learning, our method obtains significant improvement across all the datasets compared with other comparison methods. Comparing KPS loss with baseline methods (CE loss and Focal loss), we find that KPS loss significantly improves the CE loss and Focal loss. Under the setting of $\rho = 100$, for instance, KPS loss reduces the baseline methods by

more than 10% and 6% top-1 error rate on CIFAR-10-LT and CIFAR-100-LT, respectively. Compared with the loss modification methods, our proposed method outperforms these methods in each session by notable margins. Especially, compared with the most related method, namely LDAM-DRW, KPS loss reduces the top-1 error rate by 4.2% and 3.56% on CIFAR-10-LT and 2.99% and 2.59% on CIFAR-100-LT under $\rho = 100$ and 50. We can also clearly observe the superiority of KPS loss compared with two-stage methods that achieve good performance. Even though meta-learning incorporating focal loss get the best result on CIFAR-100-LT with $\rho = 50$, our proposed KPS loss outperforms it on other datasets. And our method gets the second best on CIFAR-100-LT with $\rho = 50$. Furthermore, meta-learning adopts two training stages and needs to explicitly estimate the class-conditioned distributions differences use a separate network. Our method which only modifies the loss function based on label distribution is simpler to implement compared with meta-learning.

Additionally, the following observation can be obtained from Table 3.2. The two-stage training strategies (*i.e.*, BBN and meta-learning) are also effective in most cases, since they could obtain comparable or even better results compared with other state-of-the-art methods. BBN takes two branches network to learning different distributions training samples and meta-learning uses a network to estimate the conditional distribution $P(x|y)$ from head classes and then transfer it to the tail classes. These strategies are enlightening.

3.5.4.2 Experimental Results on ImageNet-LT and iNaturalist 2018

Table 3.3 shows the results on two large-scale long-tailed datasets, *i.e.*, ImageNet-LT and iNaturalist 2018. As shown in Table 3.3, our KPS loss still outperforms baselines and the competing approaches across these two datasets, which is consistent with the observation of CIFAR-10/100-LT. Compared with the baseline methods, our method reduces the top-1 error rate on ImageNet-LT and iNaturalist 2018 by more than 6% and 10%, respectively, which is a significant performance improvement. Additionally, compared to LDAM-DRW, we have achieved notable improvements, *i.e.*, 2.48% and 2.35% improvements on ImageNet-LT and iNaturalist 2018, respectively. Even compared with the two-stage methods that

Table 3.3: Comparison on ImageNet-LT and iNaturalist 2018.

Dataset	ImageNet-LT		iNaturalist 2018
Backbone	ResNet-50	ResNeXt-50	ResNet-50
CE loss	55.49	53.35	39.89
Focal loss [36]	54.2	-	39.70
CosFace [98]	55.05	53.22	33.28
ArcFace [95]	55.46	51.51	36.14
LDAM-DRW [67]	51.20	-	32.00
CB-Focal [35]	-	-	38.88
Equalised loss [71]	52.70	50.90	38.37
LA loss [76]	**51.11**	-	31.56
BBN [50]	55.30	-	**30.38**
Meta-learning*[106]	-	-	32.45
Decoupling [12]	52.30	**50.10**	30.70
KPS loss (Ours)	**48.72**	**47.17**	**29.65**

Note: Top-1 error rates (%) are presented. The best and the second best results are shown in **underline bold** and **bold**, respectively.
* Results are obtained by incorporating CE loss.

Table 3.4: Comparison of Different Optimization Strategies on Long-tailed CIFAR Dataset.

Dataset	CIFAR-10-LT		CIFAR-100-LT	
ρ	100	50	100	50
KPS w/o GA and DRW	19.85	16.40	55.31	50.92
KPS-DRW	19.09	16.11	58.95	54.30
KPS-GA	**18.77**	**15.41**	**54.97**	**50.82**

Note: Top-1 error rates (%) are presented. The best results are shown in **underline bold**.

have achieved significant improvements, our method can surpass them.

Moreover, the two-stage fine-tuning strategies (*i.e.*, BBN and decoupling learning) also perform well. But meta-learning does not perform as well as long-tailed CIFAR on these two large-scale datasets, probably because of the loss function that meta-learning incorporates with.

3.5.5 Ablation Experiment

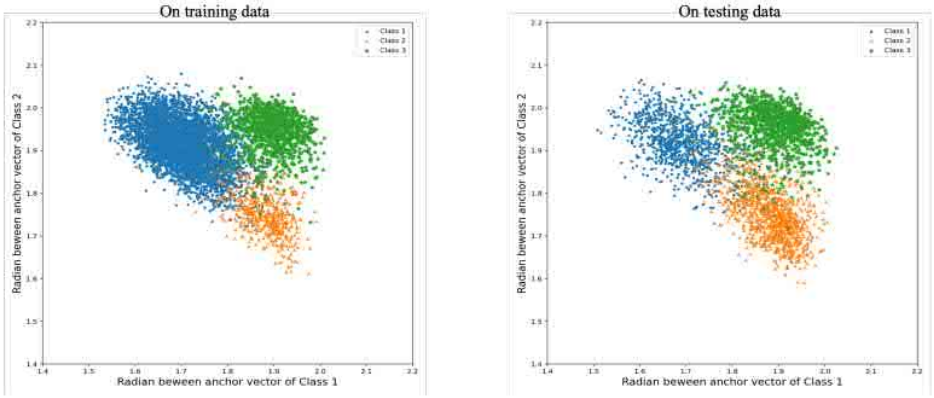
3.5.5.1 Why KPS is Effective?

In order to better explain the proposed KPS loss, we use a toy experiment to visualize feature distributions trained with different loss function. We train a ResNet-32 with 3

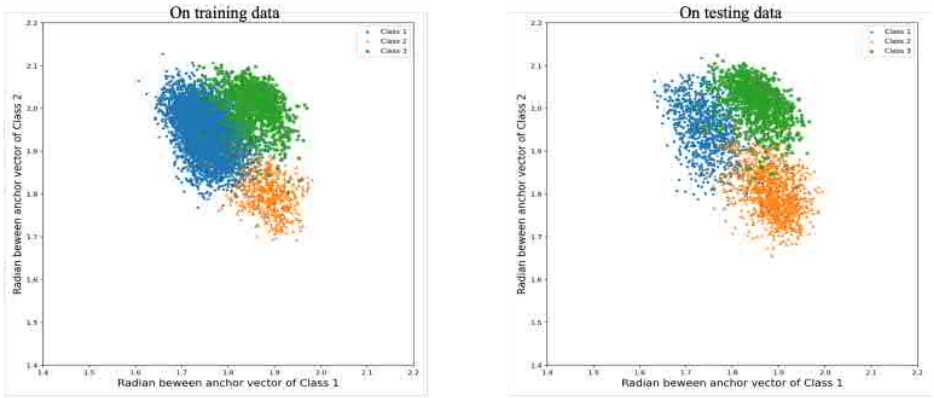
classes from CIRAR-10 with 30 epoches. The number of training samples for Class 1, Class 2 and Class 3 are 5000, 500 and 1000, respectively. The visualization is shown in Figure 3.5 and Figure 3.6. We can see that most of the points of Class 3 are far away from Class 1 and Class 2 than $\frac{\pi}{2}$ in feature space. The points of Class 3 are non-key points for class 1 and class 2. In Figure 3.5, we can observe that CE loss and focal loss have no margins, leading to difficulty in classifying each class. In Figure 3.6a, LDAM assigns margins for different classes and can make each classes more separable. But there are some key points clustered together, so that they are difficult to be classified. From Figure 3.6b, we can see that our proposed KPS loss can well cluster the samples in each class and make most points to become simple points. And the key points obtained by KPS loss become more separable.

3.5.5.2 What Is the Effectiveness of GA?

To illustrate the effectiveness of our proposed gradient adjustment optimization strategy, we explore several different optimization strategy on CIFAR-10/100-LT with different imbalance ratio. Specifically, we test our proposed loss trained with both the basic SGD without any other optimization strategy (KPS w/o GA and DRW) and with DRW [67] (KPS-DRW). The results are listed in Table 3.4. We can see that our proposed gradient adjustment strategy (KPS-GA) can yield better results than other strategies. Since Gao *et al.* [67] observed that their loss benefits a lot from DRW. We expect that employing DRW can be similarly beneficial for our KPS loss. But the experiments show that DRW even hurts the performance compared with original SGD without optimization strategy on CIFAR-100-LT. To further analyse the results, we show the per-class error rate on CIFAR-10/100-LT with $\rho = 100$ in Figure 3.7. On CIFAR-100-LT, in order to facilitate visualization, the classes are aggregated into ten groups according to their label frequency sort order. Group 1 corresponds to the top 10 most frequent classes. Group 2 comprises the second most frequent 10 classes, and so on. The most frequent 3 classes/groups in CIFAR-10/100-LT are named as Head1 to Head3. The 4 classes/groups appearing with medium frequency are named as Mid1 to Mid4 in order, and the rest 3 least frequent



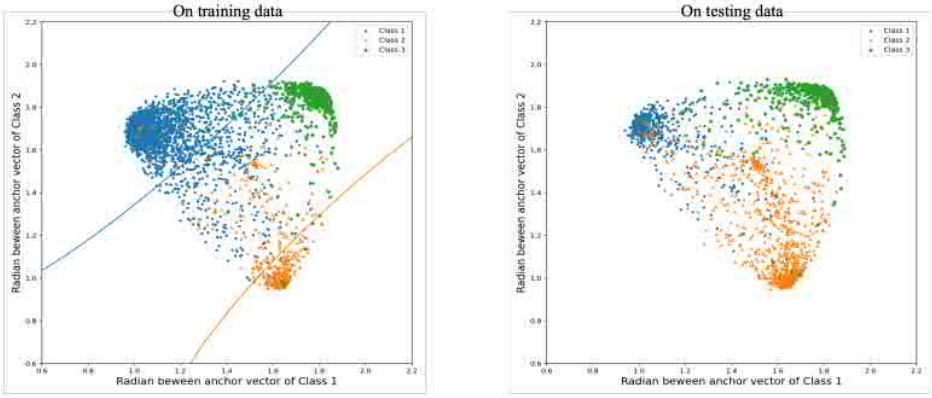
(a) CE loss



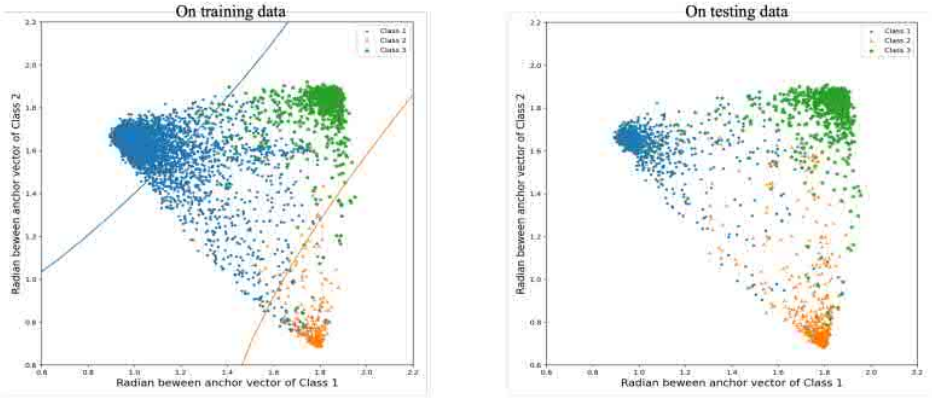
(b) focal loss

Figure 3.5: Feature distribution of baseline method. A ResNet-32 is trained on 3 classes from CIRAR-10. 5000, 500 and 1000 samples for Class 1, Class 2 and Class 3 are randomly selected, respectively.

classes/groups are named Tail1 to Tail3 in order. As shown in Figure 3.7, compared with the basic optimization strategy, DRW increases the performance on tail classes but meanwhile harms that on head classes. In contrast, the proposed GA improves the accuracy of the tail classes, and that of the head classes only slightly decreases or even improves in some cases (*e.g.*, see Head2 in CIFAR-10-LT and Head1 in CIFAR-100-LT).



(a) LDAM

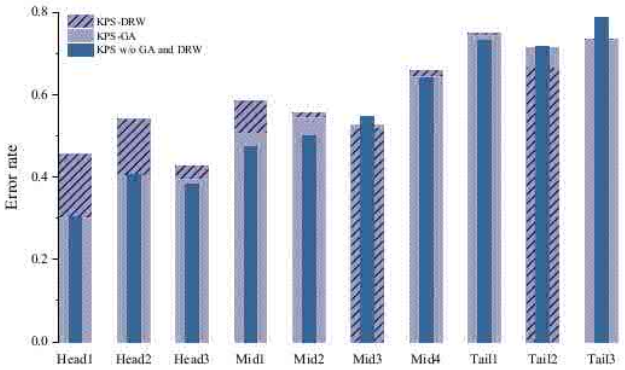


(b) KPS loss

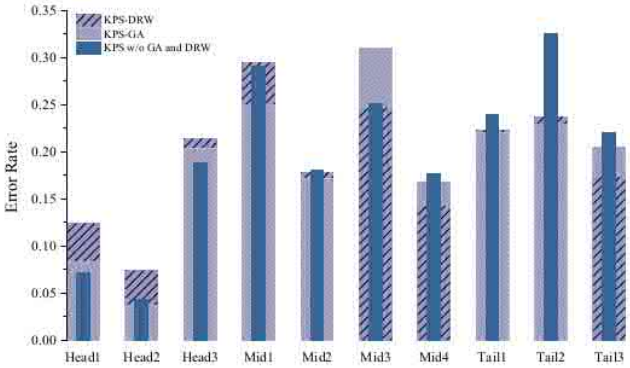
Figure 3.6: Feature distribution of LDAM and KPS. A ResNet-32 is trained on 3 classes from CIRAR-10. 5000, 500 and 1000 samples for Class 1, Class 2 and Class 3 are randomly selected, respectively.

3.5.5.3 Can KPS Loss Be Combined with Other Methods?

KPS loss can be trained end-to-end and easy to attach to other methods. We conduct the experiment of KPS combined with mixup strategy [42] and the most recent proposed two-stage method, *i.e.*, MiSLAS [44]. The results are shown in Table 3.5 and 3.6. Table 3.5 shows that mixup is an effective strategy for boosting performance. It can be found that KPS with mixup, which can be trained end-to-end, is better than MiSLAS in most cases we have tried thus far. As for the two-stage method as shown in Table 3.6, using KPS in



(a) Per-class error rates on CIFAR-10-LT



(b) Per-group error rates on CIFAR-100-LT

Figure 3.7: Per-class/group error rates obtained by different optimization strategies on CIFAR-10/100-LT with the imbalance ratio $\rho = 100$. Head classes are with low indices. Conversely, tailed-classes are with higher indices. For CIFAR-100-LT, we aggregate the classes into 10 groups.

the second stage of MiSLAS can indeed further improve the performance.

3.5.5.4 How Does KPS Loss Perform in the Head, Middle and Tail Classes?

We use CIFAR-10-LT and CIFAR-100-LT with $\rho = 100$ to present the performance on each class/group. Besides KPS loss, the per-class/group error rate with baseline methods, namely CE loss and focal loss are shown for comparison. Since CosFace, ArcFace and LDAM-DRW are the most related methods with our work, Figure 3.8 also breaks down the per-class/group error rate of these methods. Analogous to Figure 3.7, on CIFAR-100-LT, we aggregate the classes into ten groups based on their frequency-sorted order for ease

Table 3.5: Comparison of KPS Combined with Mixup.

Dataset	CIFAR-10-LT		CIFAR-100-LT		ImageNet-LT	iNaturalist 2018
Backbone	ResNet-32		ResNet-32		ResNet-50	ResNet-50
ρ	100	50	100	50	-	-
KPS	18.77	15.41	54.97	50.82	48.72	29.65
KPS with mixup	**17.84**	**15.07**	**54.66**	**49.45**	**47.05**	**28.12**

Top-1 Error rates (%) are reported.

Table 3.6: Comparison of KPS Combined with MisLAS.

Dataset	CIFAR-10-LT		CIFAR-100-LT		ImageNet-LT	iNaturalist 2018
Backbone	ResNet-32		ResNet-32		ResNet-50	ResNet-50
ρ	100	50	100	50	-	-
MiSLAS	17.94	14.84	52.62	48.28	47.89	29.43
MiSLAS with KPS	**17.26**	**14.36**	**52.28**	**47.59**	**47.74**	**29.12**

Note: Top-1 error rates (%) are reported.

of visualisation. Then, we divide the 10 classes/groups into Head, Middle and Tail according to the sample frequency. The corresponding per-class/group accuracy is shown in Figure 3.8. The following phenomena can be observed:

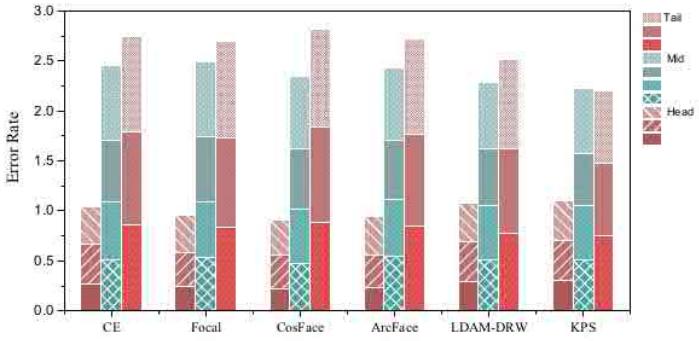
(1) The overall error rate of all methods shows the trend of Head $<$ Mid $<$ Tail. This is in line with common sense, because head classes have more sample diversity.

(2) Compared with CE loss, CosFace and ArcFace that perform well on balance data, the methods for imbalanced data, *i.e.*, Focal loss, LDAM-DRW and our KPS loss can decrease the error rate on tail classes. LDAM-DRW and KPS loss can also improve the performance on middle classes.

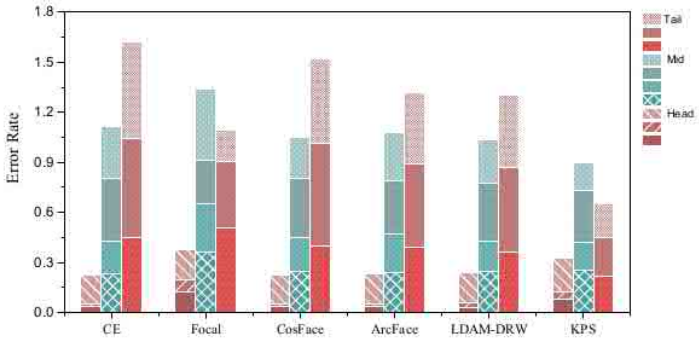
(3) The methods for imbalanced data all degrade the performance of the model in the head class to a certain extent. The reduction in the head class of KPS loss is insignificant compared to the improvement in the tail class. For example, on CIFAR-100-LT with $\rho = 100$, KPS loss improves error rate of Head1 by 1.7%, but reduces that of Tail1 by more than 10%.

3.6 Concluding Remarks

In this chapter, we have proposed the KPS loss with a gradient adjustment (GA) to solve the problem of visual classification based on training data with long-tailed distribution.



(a) Per-class error rates on CIFAR-10-LT



(b) Per-group error rates on CIFAR-100-LT

Figure 3.8: Per-class/group error rates obtained by different techniques on CIFAR-10/100-LT datasets with $\rho = 100$. Head classes are with low indices. Conversely, tailed-classes are with higher indices. For CIFAR-100-LT, the classes are aggregated into 10 groups.

We have proposed a KPS loss, which can unify several recent proposals and overcome their limitations. This KPS loss is geometrically principled and has a twofold effectiveness:

(1) Increase the margins of key points;

(2) Encourage a large relative margin between points of tail versus head classes.

To further increase the classification accuracy, we have proposed a gradient adjustment (GA) optimization strategy that can compensate the excessive punishment for tail classes though the scale parameters based on label distribution. Such adjustment encourages a large relative gradient magnitude for tail classes. The extensive experiments have demonstrated that our KPS loss with GA achieves the best performance on long-tailed benchmarks, including the most challenging dataset, *i.e.*, iNaturalist 2018.

The proposed KPS loss is able to increase the overall accuracy on each dataset compared with previous state-of-the-art methods. However, it has a limitation. As mentioned in Section 3.5.5.4, KPS loss significantly improves the performance of the middle and tail classes while sacrificing a small amount of head class accuracy. In our future work, we attempt to study the loss function that can improve the performance of head classes as well as the tail classes.

Chapter 4

Feature-Balanced Loss for Long-Tailed Visual Recognition

In this chapter, we propose a new logit adjustment method, named feature-balanced loss, to address the limitation in KPS proposed in Chapter 3. KPS loss considerably enhances the performance of the middle and tail classes, while surrendering a small percentage of head class classification accuracy.

Recent studies have tried to solve this issue by obtaining good representations from data space, but few of them pay attention to the influence of feature norm on the predicted results. We address the long-tailed problem from embedding space in this chapter and thereby propose the feature-balanced loss (FBL). Specifically, FBL encourages larger feature norm of tail classes by giving them relatively stronger stimulus. Moreover, the stimulus intensity is gradually increased in the way of curriculum learning, which improves the generalization of the tail classes, meanwhile maintaining the performance of the head classes. Extensive experiments on multiple popular long-tailed recognition benchmarks demonstrate that the feature-balanced loss achieves superior performance gains compared with the state-of-the-art methods.

4.1 Introduction

To address the issue of extreme data imbalance caused by long-tailed distribution, an intuitive way is to re-balance the model via class-balanced sampling [12], [20] or loss

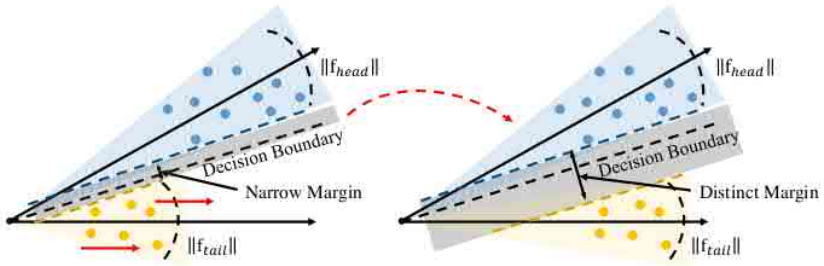


Figure 4.1: Schematic of the influence of feature norm on decision margin in embedding space. With the increase of the feature norm of the tail class samples, the margin becomes clear and the separability of the samples can be enhanced, which in turn improves the model generalization towards the samples. This feature imbalance in long-tail visual recognition tasks can be experimentally observed on manually created long-tail dataset including CIFAR-10/100-LT, ImageNet-LT, Places-LT, and natural long-tail dataset including iNaturalist 2018.

function re-weighting [89], [90]. However, these methods result in overfitting to the tail classes, which invariably inhibits the performance of the model. Most recently, Cui *et al.* [35] have proposed to re-weight the loss function or re-sample the data based on the "effective number" of each class, which has been shown empirically effective. This "effective number" strategy, on the other hand, does not truly address the issue of uneven feature distribution for long-tailed data. Cao *et al.* [67] utilized the label-distribution-aware margin (LDAM) to re-weight the loss, which can improve the generalization performance of tail classes. Nevertheless, it calculates the predicted logit through the cosine distance, which neglects the significant influence of the feature norm. In addition, these methods boost the classification performance of the tail classes at the expense of the head classes accuracy.

This chapter addresses the issue of significantly improving tail classes performance but reducing head classes accuracy. We solve this long-tailed problem from a feature norm point of view and thereby proposing the novel FBL. As shown in Figure 4.1, it can be seen that training samples with less feature norm are difficult to classify because of the unclear margins between each class. The increase of feature norm can enlarge

the margins between classes and enhance the separability of the samples. Based on this observation, we add a class-based stimulus to the predicted logit to encourage larger tail class feature norm to improve its generalization. Different from LDAM that utilizes hard margins to increase intra-class compactness, our FBL enlarges decision margin without compressing the embedding space distribution of each class. Furthermore, we adopt curriculum learning [79] strategy to gradually increase the class-based stimulus so that the network initially concentrates on the head classes, and then gradually shifts its attention to the tail classes as the training progresses. In this way, the classification accuracy of the tail classes can be improved meanwhile ensuring the model performance on the head classes. We validate the proposed FBL on five popular benchmark datasets, *i.e.*, CIFAT-10-LT, CIFAT-100-LT, ImageNet-LT, iNaturalist 2018 and Places-LT. We also conduct an additional experiment on feature norm visualization, which demonstrates that feature norm is one of the key factors for improving classification accuracy and is instructive for long-tailed study.

The main contributions of this chapter are summarized below:

(1) We propose the novel FBL for long-tailed visual recognition by adding an extra class-based stimulus to the logit. Based on the observation that the feature norm of tail classes samples always be suppressed, the proposed FBL encourages larger feature norm for tail classes, thereby improving the generalization performance of these classes.

(2) We propose to gradually increase the intensity of stimulus in the way of curriculum learning. This robust training strategy not only enhances the classification accuracy of tail classes to a large extent, but also maintains the performance of head classes.

(3) We conduct extensive experiments on commonly used long-tailed datasets, which demonstrate the superiority of the proposed method compared with the state-of-the-arts.

4.2 Related Work

The long-tailed classification methods and curriculum learning methods are briefly reviewed in this section.

4.2.1 Long-tailed Classification

Due to the prevalence of data imbalance in the real world, long-tail visual recognition has received more and more attention in the field of computer vision. This section will make an overview of the most related works, which all focus on obtaining better representation of the input images.

Loss modification methods: Loss modification aims to re-balance the importance of different classes by tuning the loss values. It addresses the class imbalance problem from two perspectives: sample-wise and class-wise. Sample-wise methods [36], [38] assign large relative weights to the difficult samples through the fine-grained parameters in the loss. For example, focal loss [36] utilizes the sample prediction hardness as the re-weighting coefficient of the loss function. However, the classification difficulty of a sample may not directly related to its corresponding class size, and thus, the sample-wise method is not capable of handling the severe imbalance data. Class-wise methods [35], [71], [89] assign the loss function with class-specific parameters that are negatively correlated to the label frequencies. For example, Cui *et al.* [35] proposed to re-weight the loss function by the "effective number" of each class instead of label frequency. Nevertheless, it does not completely alleviate the problem of biased feature distribution.

Logit adjustment methods: Logit adjustment addresses the class imbalance problem by calibrating the logit to the prior during inference or training. A typical series of approaches adjust the loss during training. Li *et al.* [114] proposed to inject per-class margins into the hinge loss and make the class boundary be closer to a dominant class. Most recently, Wang *et al.* [94] add a margin to the target logit of softmax loss. Cao *et al.* [67] have proposed label-distribution-aware margin accompanied with the deferred scheme (LDAM-DRW), which enforces tail classes to have large relative margins to increase

their classification accuracy. DisAlign [56] adaptively aligns the logit to a balanced class distribution to adjust the biased decision boundary, which can re-balance the classifier well. Another kind of method post-hoc shifts the predicted logits. For example, Menon *et al.* [76] proposed logit adjustment (LA) to post-process the logit based on the class sizes of training data. In contrast, Hong *et al.* [115] proposed LADE, which post-adjusts logits with the label frequencies of testing data, allowing the distribution of the test set to be arbitrary.

Two-stage imbalance learning: This method divides the training process into classifier learning and representation learning, which also aims to obtain good features. Decoupling the learning of classifier head and representation are the latest popular solutions to long-tail learning. Decouple [12] achieves advanced performance via properly re-balancing the classifiers. A cursory inspection of [76], [77] reveals that importing a post-process to adjust the predicted logit can improve the two-stage methods, but such a two-stage structure requires additional training of the classifier.

4.2.2 Curriculum Learning

Curriculum learning was first proposed by Bengio *et al.* [79]. It draws on the learning process that humans and animals generally follow in an order from easy to difficult. Curriculum learning advocates the model to start learning from easy samples and gradually shift the model's attention to complex samples. This strategy helps the model to speed up the training and obtain better generalization. Utilizing curriculum learning to control the order in which samples are presented to neural networks during training [116]–[118] is attracting more attention with the development of deep learning and its rising importance in many applications. For example, Pentina *et al.* [119] applied curriculum learning to find the optimal order of subsequent tasks to be learned for multi-task learning. Liu *et al.* [120] proposed a curriculum learning based framework to address the issue of noisy and uneven-quality corpora in question answering. There is also a lot of work [30], [50], [121] applying curriculum learning to imbalanced data classification.

4.3 Proposed Method: Feature-balanced Loss

To mitigate the training bias towards the head classes caused by long-tailed data, we propose the FBL as a more powerful supervised signal for optimizing CNNs.

4.3.1 Motivation

We revisit the original softmax loss function utilized for multi-class classification first. The softmax loss function for a given sample x is:

$$\mathcal{L}_{\text{softmax}}(x) = -\log \frac{e^{z_y}}{\sum_j e^{z_j}}. \tag{4.1}$$

The gradients of $\mathcal{L}_{\text{softmax}}$ w.r.t.z_i is:

$$\frac{\partial \mathcal{L}_{\text{softmax}}}{\partial z_i} = \begin{cases} p_i - 1, & i = y \\ p_i, & i \neq y \end{cases}, \tag{4.2}$$

where $p_i = \frac{e^{z_i}}{\sum_j e^{z_j}}$. In backward propagation, the gradients of the target class are negative, and those of the non-target classes are positive. Thus, the training samples punish the non-target class weights $\mathbf{w}_i$ $(i \neq y)$ by p_i. The weights of tail classes which have fewer training instances always receive punishment signals. As a result, the weight norm of the classifier for tail classes is always reduced. Therefore, we obtain the following properties:

Property 1. The weight norm $\|\mathbf{w}_i\|$ of the classifier for class i are correlated with the class size n_i.

In addition, we introduce additional property of softmax loss that was found by Yuan *et al.* [122]:

Property 2. By fixing the weight vector and direction of feature vectors, softmax loss is a function that monotonically decreases with the increasing of feature $L2$-norm when features are correctly classified.

Property 1 indicates that the target logit $z_y = \mathbf{w}_y^T \mathbf{f}$ of tail class is usually suppressed because of the relatively small $\mathbf{w}_y^T$. Encouraging a larger tail class norm can alleviate its over-suppressed logit. Meanwhile, Property 2 shows that feature norm is an important

factor to achieve a lower loss, which can provide power to make the features to be more separable. In addition, Figure 4.1 geometrically shows that the margin becomes clear as the norm of the tail sample feature increases. As a result, the model generalization towards tail class samples can be improved. Therefore, the benefits of encouraging larger feature norm for tail classes are three-fold:

(1) Increase the over-suppressed tail classes logits;

(2) Improve the separability of the sample features;

(3) Enlarge the margin among classes in embedding space.

In this way, the model bias towards the head classes can be effectively diminished.

4.3.2 Feature Norm Balancing

As analyzed in Section 4.3.1, we try to encourage large feature norm of tail classes. To stimulate the large feature norm, we can add an additional constraint item to the original cross-entropy loss:

$$L' = -\log \frac{e^{z_y}}{\sum_i e^{z_i}} + \alpha \frac{\lambda_y}{\|\mathbf{f}\|}, \tag{4.3}$$

where α is the parameter used to adjust the strength of the constraint, and λ_y controls the stimulus intensity towards different classes. The constraint item $\alpha \frac{\lambda_y}{\|\mathbf{f}\|}$ can regularize the length of feature norm of the target class to make it large.

Since Property 1 in Section 4.3.1 states that the weight norm of classifier for tail classes is usually suppressed, the logits of tail classes will be unfairly reduced. To diminish this bias, we can encourage large feature norm for tail classes and thus assign them stronger stimulation. λ_j controls the intensity of the stimulus, which should be strong for tail classes and weak for head classes. Therefore, λ_y is negatively correlated with the number of samples in class y. We set λ as:

$$\lambda_j = \log n_{max} - \log n_j, \tag{4.4}$$

so that it is zero for the most frequent class and is much stronger for tail classes.

For the sake of analysis of the loss function, we write Equation (4.3) as:

$$\begin{aligned} L' &= -\log \frac{e^{z_y}}{\sum_j e^{z_j}} + \log e^{\frac{\lambda_y}{\|\mathbf{f}\|}} \\ &= -\log \frac{e^{z_y - \frac{\lambda_y}{\|\mathbf{f}\|}}}{\sum_j e^{z_j}} \\ &= -\log p_y \end{aligned}, \tag{4.5}$$

where $p_y = \frac{e^{z_y - \frac{\lambda_y}{\|\mathbf{f}\|}}}{\sum_j e^{z_j}}$. As the sum of the probabilities of all classes obtained by Equation (4.5) is not equal to 1, *i.e.*, $\sum_{y=1}^{C} p_y \neq 1$, which does not satisfy the properties of probability. We therefore further modify the logit to ensure that the total predicted probabilities of all classes are equal to 1. The feature-balanced logit z_j^b of class j is introduced and is expressed as:

$$z_j^b = z_j - \alpha \frac{\lambda_j}{\|\mathbf{f}\|}. \tag{4.6}$$

In this way, $p_y = \frac{e^{z_y - \alpha \frac{\lambda_y}{\|\mathbf{f}\|}}}{\sum_j e^{z_j - \alpha \frac{\lambda_j}{\|\mathbf{f}\|}}}$. It satisfies the properties of probability that $\sum_{y=1}^{C} p_y = 1$. At the same time, this modified logit can encourage larger feature norms of tail classes to achieve the purpose of balancing features of different classes.

4.3.3 FBL with Curriculum Learning

Furthermore, in Equation (4.5), the stronger the constraint (*i.e.*, $\frac{\lambda_j}{\|\mathbf{f}\|}$) on the feature is, the more the model focuses on the tail classes. We can adopt the idea of curriculum learning [79], which makes the model initially focus on easy samples (*i.e.*, head classes), and then gradually shift to learning difficult samples (*i.e.*, tail classes). To achieve this purpose, we can choose the learning strategy that gradually increases α as the training progresses. Therefore, α is set as $\alpha(t)$ which is related to the training epoch t. We empirically select the parabolic increase learning strategy, which is expressed as:

$$\alpha(t) \propto (\frac{t}{T})^2, \tag{4.7}$$

where t is the current training epoch and T is the total epochs for training. Section 4.4.4.1 also provides experimental results for different learning strategies.

Algorithm 4.1: FBL with curriculum learning

Input: Training dataset $\mathcal{T}$
Output: Predicted labels
Initialize the CNN model $\phi((x, y); \omega)$ randomly, where ω is the parameter of the model;
for $t = 1$ *to* T **do**
  Sample mini-batch training samples $\mathcal{B}$ from the long-tailed data $\mathcal{T}$ with batch size of b;
  Calculate the constraint strength parameter α by Equation (4.7): $\alpha \leftarrow \alpha(t)$;
  Calculate the stimulus intensity parameter λ_j by Equation (4.4): $\lambda_j \leftarrow \log n_{max} - \log n_j$;
  Calculate the loss by Equation (4.8): $\mathcal{L}((x, y); \omega) = \frac{1}{b} \sum_{(x,y)\in\mathcal{B}} L_{\text{FBL}}(x, y)$;
  Update model parameters: $\omega \leftarrow \omega - \alpha' \nabla_\omega \mathcal{L}((x, y); \omega)$;
end

The final loss function $\mathcal{L}_{\text{FBL}}$ is expressed as:

$$\begin{aligned} L_{\text{FBL}} &= -\frac{1}{N} \sum_i \log \frac{e^{z_y^b}}{\sum_j e^{z_j^b}} \\ &= -\frac{1}{N} \sum_i \log \frac{e^{z_y - \alpha(t)\frac{\lambda_y}{\|\mathbf{f}\|}}}{\sum_j e^{z_j} - \alpha(t)\frac{\lambda_j}{\|\mathbf{f}\|}}. \end{aligned} \tag{4.8}$$

This loss function is named as FBL–feature-balanced loss, because it balances the logit of different classes based on feature norm. The algorithm of our proposed method is summarized in Algorithm 4.1.

4.3.4 Comparison with Previous Methods

Some other methods, for example, logit adjustment (LA) [76], balanced meta-softmax (BALMS) [31] and seesaw loss [74] also balance the features distribution. The section mathematically compares their differences.

The loss function of logit adjustment is

$$L_{LA} = -\log \frac{e^{z_y + \log n_y^\tau}}{\sum_{j=1}^{C} e^{z_j + \log n_j^\tau}}, \tag{4.9}$$

where $\tau > 0$ is a constant.

The expression of the BALMS loss function is

$$L_{BALMS} = -\log \frac{e^{z_y+\log n_y}}{\sum_{j=1}^{C} e^{z_j+\log n_j}}. \tag{4.10}$$

The loss function of seesaw loss is expressed as When $N_y < N_j$ $(j \neq y)$,

$$L_{seesaw} = L_{CE} \tag{4.11}$$

When $N_y > N_j$ $(j \neq y)$,

$$\begin{aligned} L_{seesaw} &= -\log \frac{e^{z_y}}{\sum_{j\neq i}^{C} e^{\log \frac{\sigma_j{}^q N_j{}^p}{\sigma_i{}^q N_i{}^p}} e^{z_j} + e^{z_i}} \\ &= -\log \frac{e^{z_y}}{\sum_{j\neq i}^{C} e^{z_j+\log \sigma_j{}^q N_j{}^p - \log \sigma_i{}^q N_i{}^p} + e^{z_i}} \\ &= -\log \frac{e^{z_y+\log \sigma_y{}^q N_y{}^p}}{\sum_{j=1}^{C} e^{z_j+\log \sigma_j{}^q N_j{}^p}} \\ &= -\log \frac{e^{z_y+\nu_y}}{\sum_{j=1}^{C} e^{z_j+\nu_j}} \end{aligned}, \tag{4.12}$$

where $\sigma_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}}$ is the confidence of class i. $\nu_i = \log \sigma_i{}^q N_i{}^p$ is correlated with the size and confidence of class i. Seesaw loss accumulates instance number at each iteration during training. When $N_y > N_j$, it is similar with BALMS.

From Equation (4.9), Equation (4.11) and Equation (4.12), we can see that BALMS, LA and seesaw loss aim to adjust the logit and obtain an unbiased extension of softmax. They solve the long-tailed visual problem from the perspective of minimizing the generalization bound. The adjustment of logit by these methods will cause a certain damage to the classification accuracy of the head classes. Differently, we utilize the feature norm to balance the predicted logit, which does not change the margins among head classes and is neglected by the aforementioned methods.

Table 4.1: Summary of Basic Setting for FBL

Dataset	CIFAR-10/100-LT	ImageNet-LT	iNaturalist 2018	Places-LT
Backbone	ResNet-32	ResNet-50	ResNet-50	ResNet-152
Bini-bactch size	128	512	512	512
Initial lr	0.1	0.2	0.2	0.2
Weight decay	2×10^{-4}	2×10^{-4}	1×10^{-4}	5×10^{-4}
Training epochs	200	200	180	90
lr rate schedule	multi-step	multi-step	multi-step	multi-step
Decay ratio of lr	0.01	0.01	0.1	0.1
Decay epochs of lr	{160,180}	{160,180}	{120,160}	{60,80}

4.4 Experiments

4.4.1 Implementation Details

We use Pytorch [112] to implement and train all the backbones with stochastic gradient descent with momentum.

4.4.1.1 Backbone

Following the protocol of Cui *et al.* [35], ResNet-32 is adopted as the backbone for all CIFAR-10/100-LT datasets. For ImageNet-LT and iNat 2018, ResNet-50 is applied. For Places-LT, we follow Liu *et al.* [82] and start from a ResNet-152 pre-trained on the original balanced version of ImageNet. Except for ResNet-152, all the backbones are trained from scratch.

4.4.1.2 Training Details

For CIFAR-10/100-LT, we train the backbone with 200 epochs and batch size of 64. The initial learning rate (lr) is set to 0.1, and we anneal lr by 100 at the 160-th and 180-th epoch, respectively. For the three large-scale datasets, backbone is trained with 180 epochs, batch size of 512, and initial $lr = 0.2$. We divide lr by 10 at 120-th and 160-th epochs.

The training details of FBL are summarized in Table 4.1.

4.4.2 Comparison Methods

The vanilla training with basic cross-entropy (CE) loss is chosen as the baseline method. We compare with the state-of-the-art from the most related fields. That is, the logit

Table 4.2: Comparison Results on CIFAR-10/100-LT.

Dataset	CIFAR-10-LT		CIFAR-100-LT	
Backbone Net	ResNet-32			
ρ	100	50	100	50
CE loss (baseline)	71.07	75.31	39.43	44.20
LDAM-DRW [67] (*NeurIPS* 2019)	77.03	81.03	42.04	47.62
BBN [50] (*CVPR* 2020)	79.82	81.18	42.56	47.02
LA [76](*ICLR* 2021)	80.92	-	43.89	-
FBL (ours)	**82.46**	**84.30**	**45.22**	**50.65**

Note: Top-1 accuracy (%) is reported. The best results are shown in **underline bold**.

Table 4.3: Comparison results on ImageNet-LT, iNaturalist 2018 and Places-LT.

Dataset	ImageNet-LT	iNat 2018	Places-LT
Backbone Net	ResNet-50	ResNet-50	ResNet-152
CE loss (baseline)	44.51	63.80	27.13
LDAM-DRW [67] (*NeurIPS* 2019)	48.80	68.00	-
Decoupling [12] (*ICLR* 2020)	47.70	69.49	37.62
LA [76] (*ICLR* 2021)	50.44	66.36	-
FBL (ours)	**50.70**	**69.90**	**38.66**

Note: Top-1 accuracy (%) is reported. The best results are shown in **underline bold**.

modification methods which include LDAM-DRW [67] and LA [76], the most recently proposed two-stage method–BBN [50] on the small-scale datasets (*i.e.*, CIFAR-10/100-LT) and decoupling on the large-scale datasets (*i.e.*, imageNet-LT, iNat2018 and Places-LT).

4.4.3 Long-Tailed Recognition Results

4.4.3.1 Results on Small-scale Datasets

We conduct the comparison experiments on CIFAR-10/100-LT with $IF = \{100, 50\}$. Table 4.2 summarizes the top-1 accuracy. Our FBL outperforms prior arts by noticeable margins across all the datasets compared with other competing methods. For example, FBL outperforms the state-of-the-art method – LA by 1.54% and 1.33% with $IF = 100$ on CIFAR-10-LT and CIFAR-100-LT, respectively.

4.4.3.2 Results on Large-scale Datasets

FBL yields good performance on all large-scale datasets, which is consistent with the results on CIFAR-10/100-LT. The results are shown in Table 4.3. The proposed FBL that

Table 4.4: Ablation Experiment of Different Learning Strategy.

$\alpha(t)$	Representation	Accuracy(%)
Linear decrease	$1-\frac{t}{T}$	75.97
Linear increase	$\frac{t}{T}$	81.67
Sine increase	$\sin(\frac{t}{T}\cdot\frac{\pi}{2})$	81.22
Cosine increase	$1-\cos(\frac{t}{T}\cdot\frac{\pi}{2})$	80.79
Parabolic increase	$(\frac{t}{T})^2$	**82.46**

Note: The experiments are conducted on CIFAR-10-LT with $\rho = 100$. Top-1 accuracy (%) is reported.

61 can be trained end-to-end not only achieves better results than LA, but also is superior to the two-stage method, *i.e.*, LDAM-DRW and decoupling. For example, on ImageNet-LT, FBL outperforms LDAM-DRW and decoupling by 1.90% and 3.00%, respectively.

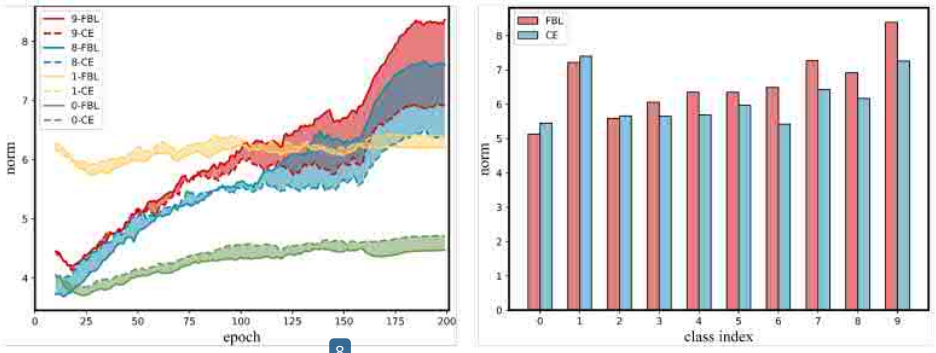
4.4.4 Ablation Study

4.4.4.1 Adjustment Strategies for Curriculum Learning

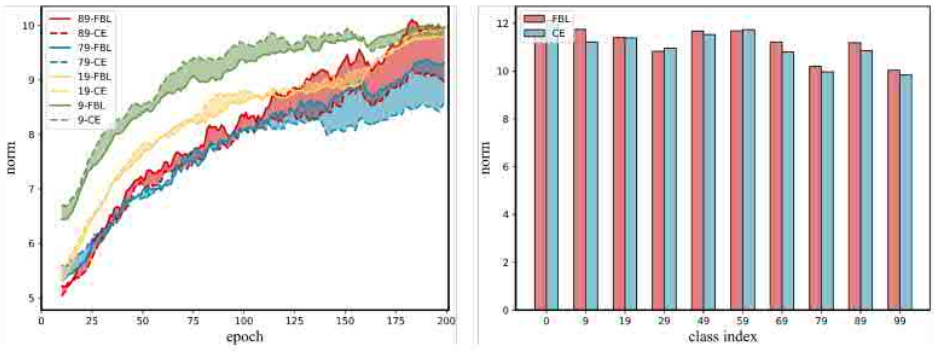
140 We conduct an ablation study to illustrate the effectiveness of different learning adjustment strategies adopted by $\alpha(t)$. Table 4.4 summarizes their performance. 2 It can be seen that the classification accuracy of the linear decrease strategy is 75.97%, which is only higher than that of the baseline method (71.07%). It is not as competitive as other learning strategies, because it makes the DNN model focus on hard samples (*i.e.*, tail classes) first. As the training progresses, the network gradually forgets what it has previously learned. Therefore, there is basically no improvement in the performance of the tail classes. Other strategies that increase α with the training epoch t gradually shift the network's attention from the head classes to the tail, which can avoid forgetting the tail classes and improve the overall performance.

4.4.4.2 Feature-balanced Results

To further validate the effects of the proposed FBL, especially the tail classes, we perform the visualization of feature norm (*i.e.*, $\|\mathbf{f}\|$) changing with respect to training epochs and feature norm distribution of classes over the 2 test set on CIFAR-10/100-LT. The results are



(a) On CIFAR-10-LT with $\rho = 100$

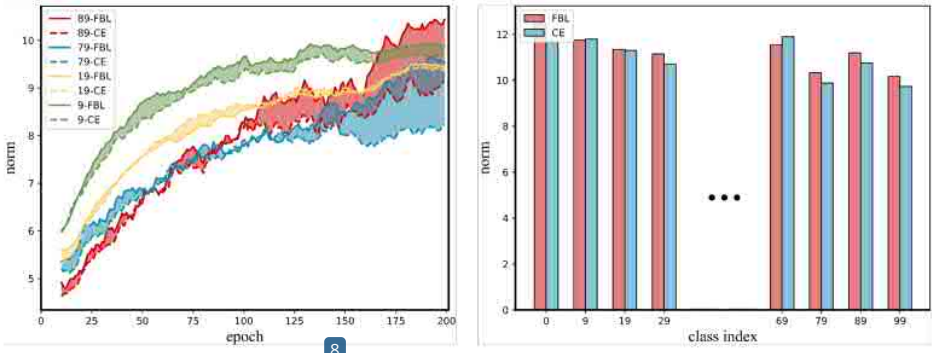


(b) On CIFAR-10-LT with $\rho = 50$

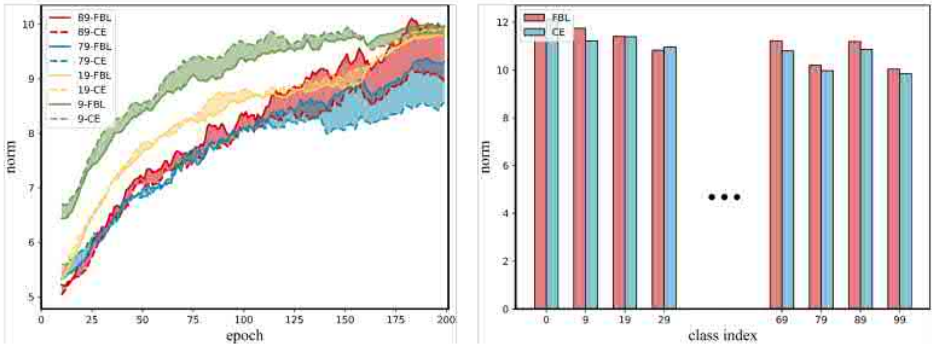
Figure 4.2: Feature norm changing on *head classes* (class index-$\{0, 1\}$) and *tail classes* (class index-$\{8, 9\}$) with respect to training epochs (left) and the feature norm distribution of classes over test dataset (right) on CIFAR-10-LT with $\rho = 100$ (a) and 50 (b).

shown in Figure 4.2 and Figure 4.3. The corresponding per-class accuracy is presented in Table 4.5 and 4.6, respectively. The following phenomena can be seen:

(1) The capability of the model to learn from different classes of samples is diverse. Specifically, in Figure 4.2, the feature norms of head classes (class index-$\{0, 1\}$) reach stable in very early training epochs due to enough training samples. Differently, on CIFAR-100-LT (as shown in Figure 4.3), the feature norms of all classes including head classes (class index-$\{9, 19\}$) and tail classes (class index-$\{79, 89\}$) are constantly changing as the epoch increases, which have a similar phenomenon to the tail classes in CIFAR-10-LT (class index-$\{8, 9\}$ in Figure 4.2), since they all suffer from insufficient training samples.



(a) On CIFAR-100-LT with $\rho = 100$



(b) On CIFAR-100-LT with $\rho = 50$

Figure 4.3: Feature norm changing on *head classes* (class index-$\{9, 19\}$) and *tail classes* (class index-$\{79, 89\}$) with respect to training epochs (left) and the feature norm distribution of classes over test dataset (right) on CIFAR-100 with $\rho = 100$ (a) and 50 (b).

(2) Compared with CE loss, our FBL encourages larger feature norms of tail class samples to eliminate representation bias towards head classes. The area $S^{\star}_{area}$(class index) enclosed by the curve of CE loss and FBL becomes larger as the number of class samples decreases, e.g. $S^{tail}_{area}(9) > S^{tail}_{area}(8) > S^{head}_{area}(1) > S^{head}_{area}(0)$ in Figure 4.2a, which is in line with our motivation.

These observations not only justify our intuition about the influence of feature norm on decision margin, but also offer a new way to investigate long-tailed visual recognition.

Table 4.5: Per-class Accuracy (%) of Test Set on CIFAR-10-LT.

Class index	0	1	2	3	4	5	6	7	8	9
ρ	100									
CE loss	**91.0**	**98.2**	**83.2**	72.5	78.8	65.1	68.8	59.5	49.0	44.6
FBL	88.1	94.7	81.9	**73.0**	**83.6**	**75.1**	**86.3**	**77.3**	**82.7**	**81.9**
ρ	50									
CE loss	**84.5**	**95.8**	68.5	**74.6**	81.1	72.7	82.9	67.5	59.1	66.4
FBL	83.7	92.1	**81.7**	73.9	**85.0**	**76.1**	**87.7**	**85.0**	**88.5**	**89.3**

Table 4.6: Per-class Accuracy (%) of Test Set on CIFAR-100-LT.

Class index	0	9	19	29	···	69	79	89	99
ρ	100								
CE loss	**89.0**	72.0	**59.0**	**48.0**	···	45.0	12.0	3.0	2.0
FBL	86.0	**77.0**	54.0	45.0	···	**60.0**	**27.0**	**22.0**	**8.0**
ρ	50								
CE loss	**88.0**	**79.0**	53.0	49.0	···	53.0	10.0	19.0	13.0
FBL	87.0	77.0	**56.0**	**57.0**	···	**62.0**	**48.0**	**38.0**	**17.0**

4.5 Concluding Remarks

In this chapter, we have proposed a novel FBL to address the long-tailed classification problem from the perspective of feature norm.

FBL encourages larger feature norm of tail classes by adding relatively stronger stimulus to the logit of tail classes, which can mitigate the representation bias towards head classes in the feature space. In addition, a curriculum learning strategy has been adopted to gradually increase the stimulus in training, which can keep the good accuracy of the model for the head classes and improve the performance of the tail classes. FBL allows CNNs to be trained end-to-end without the risk of a performance drop from head classes. Extensive experiments have demonstrated the superiority of the proposed FBL and the analysis of visualization verifies our motivation.

FBL can be trained end to end. However, compared to recently proposed two-stage methods, our FBL does not competitive enough. Contrastive learning [59] and DisAlign [56] are two kinds of two-stage methods. Although they inevitably increase model or training complexity, they can achieve competitive promotion, as Table 4.7 shown. From Table 4.5 and Table 4.6, we can make the observation that the performance of the model in the tail

Table 4.7: Comparison Results with Recently Proposed Two-stage Methods.

(a) On small-scale datasets

Dataset	CIFAR-10-LT		CIFAR-100-LT	
Backbone Net	ResNet-32			
ρ	100	50	100	50
Contrastive learning* [59] (*CVPR* 2021)	81.40	**85.36**	**46.72**	**51.79**
FBL (ours)	**82.46**	84.30	45.22	50.65

(b) On large-scale datasets

Dataset	ImageNet-LT	iNaturalist 2018	Places-LT
Backbone Net	ResNet-50	ResNet-50	ResNet-152
DisAlign* [56] (*CVPR* 2021)	**52.91**	**70.06**	**39.30**
FBL (ours)	50.70	69.90	38.66

Note: Top-1 accuracy (%) is reported. The better results are shown in **underline bold**.
* Results are quoted from the corresponding reference.

classes still has a lot of room for improvement, especially for CIFAR-100-LT. Our future work attempts to further improve the performance of the tail classes while ensuring the accuracy of the head classes.

Chapter 5

Long-tailed Visual Recognition via Gaussian Clouded Logit Adjustment

This chapter proposes a new logit adjustment method called Gaussian clouded logit (GCL) to address the limitation in FBL and therefore better handling the long-tailed visual recognition problem.

We observe that vanilla training on long-tailed data with cross-entropy loss makes the instance-rich head classes severely squeeze the spatial distribution of the tail classes, which leads to difficulty in classifying tail class samples. Furthermore, the original cross-entropy loss can only propagate gradient short-lively because the gradient in softmax form rapidly approaches zero as the logit difference increases. This phenomenon is called softmax saturation. It is unfavorable for training on balanced data, but can be utilized to adjust the validity of the samples in long-tailed data, thereby solving the distorted embedding space of long-tailed problems. To this end, this chapter proposes the Gaussian clouded logit adjustment by Gaussian perturbation of different class logits with varied amplitude. We define the amplitude of perturbation as cloud size and set relatively large cloud sizes to tail classes. The large cloud size can reduce the softmax saturation and thereby making tail class samples more active as well as enlarging the embedding space. To alleviate the bias in a classifier, we therefore propose the class-based effective number sampling strategy with classifier re-training. Extensive experiments on benchmark datasets validate the superior performance of the proposed method.

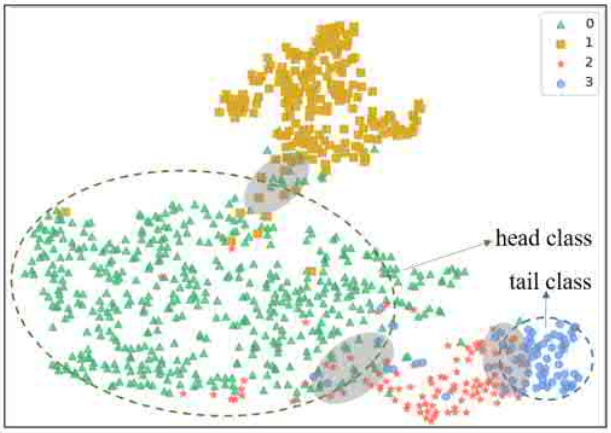


Figure 5.1: t-SNE visualization of the distorted embedding space. (Color for the best view.) The embeddings are calculated with ResNet-32 on a subset with four classes of CIFAR-10-LT. We randomly select four classes with the training numbers 500, 200, 100, and 50, respectively. The distributions of the head and tail classes are severally uneven. And the softmax saturation leads to insufficient training so that there are obscure regions (the gray area) between different classes.

5.1 Introduction

Naive learning on long-tailed data is prone to undesirable bias towards the head classes which occupy the majority of the training samples [11]. Since tail classes have few training samples that cannot cover the real distribution in embedding space, their spatial span is severely compressed by head classes. In addition, a vast number of head class samples generate overwhelming discouraging gradients for tail classes. Therefore, the learning of a classifier is biased towards the head classes. As a result, directly training on long-tailed data brings two key problems: (1) the distorted embedding space, and (2) the biased classifier.

In the literature, most of the recently proposed approaches focus on addressing the second problem only, *i.e.*, the biased classifier. For example, Menon *et al.* [76] and Hong *et al.* [115] applied post-adjust strategy to the trained model to calibrate the class boundary. Nevertheless, the distorted embedding cannot be adjusted with the post-hoc calibration, which is not conducive to further improving the model performance. Most recently, the two-stage decoupling methods [12], [50], [55], [56], [67] have been proposed to obtain good embeddings in the first stage and then re-balance the classifier in the second stage.

These methods obtain the representation by cross-entropy (CE) loss, which, however, leads to a severely uneven distributed embedding space. We implement a toy experiment to illustrate the distortion of the embedding space as shown in Figure 5.1, where t-SNE [123] is utilized to visualize the features of a long-tailed subset from CIFAR-10 dataset. We can observe that the tail class occupies a much small spatial span than the head class. This is because the tail class with fewer samples cannot cover the ground truth distribution. Moreover, Figure 5.1 also shows that there are obscure regions (*i.e.*, the grey area) between different classes. Softmax saturation [124] is one of the factors of these obscure regions because it leads to insufficient training. These obscure regions have a severe effect on the tail classes but little on the head classes. Since tail class samples clustered around the class boundary aggravate their spatial squeezing, while the head class samples with enough variety can already cover the true distribution.

Softmax saturation refers to the inopportune early gradients vanishing produced by the softmax [124], [125], which weakens the validity of training samples and impedes model training. However, from another perspective, the seemingly harmful softmax saturation has the ability to balance the valid samples of different classes and thus help calibrate the distortion of embedding space. Specifically, we disturb the logit of different classes with different amplitudes. We name the disturbed logit as Gaussian clouded logit and the amplitude of the disturbance as cloud size, because we set the disturbance to a Gaussian distribution. The tail classes have few training samples and thus the training samples of them should be more valid. We therefore disturb the logit of tail classes with large relative cloud sizes to reduce the softmax saturation. In this way, tail class samples can provide more gradients without overfitting and thus indirectly affect their embedding space. In addition, a large cloud size of the tail class logit corresponds to the large cloud size on feature in the direction of the class anchor. Therefore, tail classes can have large margins towards the class boundary, so as to alleviate the severe uneven distribution between the head and tail classes. Conversely, the head classes are set to small cloud sizes, so that they can be automatically filtered out during training. Eventually, as shown in Figure 5.2, the

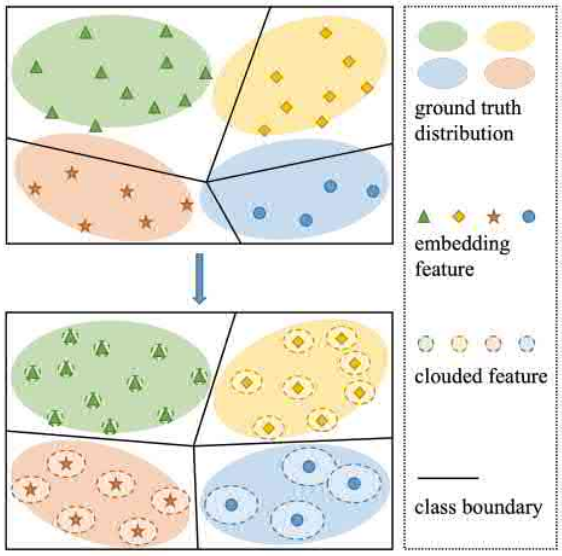


Figure 5.2: An overview of GCL. (Color for the best view.) The tail class logit is assigned to a larger sample cloud size than the head class, which corresponds to a large relative cloud size of the feature in the direction of the tail class anchor. In this way, the distortion of the embedding space can be calibrated well.

tail class samples can be pushed more away from the class boundary so as the distortion of the embedding space can be calibrated.

From another perspective, the corresponding large cloud size on the feature can be seen as a cluster of virtual samples similar to the given training example. The physical meaning of these virtual samples is meaningful, because we can get inspiration from the nature of human cognition. Human beings are capable of inferring the extension of a whole category from one example [126]. Therefore, one training sample can actually represent a set of similar samples. We assume that the distribution of these samples is Gaussian. Then, we can sample a series of virtual samples around the input image. The tail classes lack instances of different types. Hence, the sample cloud sizes of tail classes are relatively large in order to enrich the intra-class diversity.

To address the biased classifier, we re-balance the training data with a class-wise sampling strategy. As training with GCL makes the validity of different classes vary, the so-called "effectiveness" [35] of them are different. Existing class-wise balanced sampling

strategies will lead to excessive training of tail classes for GCL. We thereby propose the class-based effective number (CBEN) sampling strategy, which is based on sample validity and label frequencies to re-balance the classifier. This simple but effective sampling strategy helps mitigate the classifier bias towards the head classes and further boost the performance of GCL.

Extensive experiments on multiple commonly used long-tailed recognition benchmark datasets demonstrate that the proposed GCL surpasses the recently proposed counterparts. In summary, the key contributions of our work are three-fold:

(1) We propose the GCL adjustment loss function, which utilizes softmax saturation to balance the sample validity of different classes. An evenly distributed embedding can be obtained with the proposed GCL.

(2) We propose a simple but effective class-based effective number (CBEN) sampling strategy for re-balancing the classifier to avoid repeat training of tail classes. This sampling strategy can further boost the performance of GCL.

(3) Extensive experiments on popular long-tailed datasets demonstrate that the proposed method outperforms the state-of-the-art counterparts.

5.2 Related Works

Long tail classification is one of the long-standing research problems in machine learning, the key issue of which is data imbalance. There are several approaches proposed to address this problem, which can be roughly divided into two regimes: one-stage methods and two-stage methods. Besides, there are still some other methods, for example, transfer learning and meta-learning. This section briefly introduces the most related regimes.

5.2.1 One-stage Methods

Data re-balancing is widely used one-stage method, which includes two strategies: re-sampling and re-weighting. Re-sampling is typically accomplished by randomly over-sampling [28], [127]–[129] the tail class samples or under-sampling [19], [130], [131] the head class samples. Over-sampling leads to overfitting to the tail classes [13] and

Drummond *et al.* [130] have proven that under-sampling yields better results than over-sampling. However, under-sampling incurs the risk of discarding important data, which will lead to degradation of the model generalization ability. Re-weighting methods are another alternative prominent data re-balancing methods. Sample-wise re-weighting methods [36], [38] attempt to make the model pay more attention to the difficult samples by introducing fine-gained coefficients in the loss for imbalanced learning. Class-wise re-weighting methods [35], [67], [71], [90], [132] assign the standard cross-entropy loss with category-specific parameters that are inversely proportional to the class frequencies. However, the classification difficulty of a sample is not directly related to its corresponding class size. And another side effect of assigning higher weights to difficult samples/tail classes is overly focusing on harmful samples (*e.g.*, noisy data or mislabeled data) [133]. As analyzed above, data re-balancing methods cannot produce new information about the head class and thus suffering from overfitting to tail classes or discarding information of head class.

Data synthesis and augmentation is another remedy, which generates new samples for the tail classes. Traditional data augmentation including rotation, flipping and color jittering directly manipulates the input images in data space, which is widely employed to train DNNs [1], [134], [135]. Recently, many approaches that synthesis new samples from feature space have been proposed to complement the traditional augmentation technologies. For example, Liu *et al.* [52] transfer the intra-class angular variance of head classes to enrich the diversity of samples in tail classes. Shuang *et al.* [136] augment the tail classes by learning transformed semantic directions with meta-learning. Zang *et al.* [32] augment all the data with an online Gaussian prior from previously observed real features and incorporate with an adaptive feature sampling scheme adjusting by a balanced meta validation set. However, the noise in the augmented data may mislead the model.

5.2.2 Two-stage Methods

Many recent works have focused on improving the long-tailed visual recognition performance by decoupling the representation and classifier. Most recently, Cao *et al.* [67]

proposed the LDAM to learning features in stage I and adopted the deferred re-weighting (DRW) to fine-tune the decision boundary in stage II. It significantly improves the long-tailed prediction accuracy compared with one-stage methods, but the theoretical explanation of DRW is not clear.

After that, Kang *et al.* [12] precisely pointed out that the learning process of representation and classifier can be decoupled into two separate stages. The representation learning is conducted on the original long-tailed data in stage I and the classifier learning is performed on class-balanced re-sampling data in stage II. A lot of works [44], [55], [56], [137] have further refined this strategy. For example, Zhang *et al.* [56] proposed an adaptive calibration function to calibrate the predicted logits of different classes into a balanced class prior in stage II. Zhong *et al.* [44] proposed label distribution based soft label to deal with different degrees of over-confidence for classes and can improve the classifier learning in stage II.

Another alternative direction is proposed by Zhou *et al.* [50], which splits the network structure into two branches that focus on learning the representation of head and tail classes, respectively. This method incorporates feature mixup [49] into a cumulative learning strategy and also achieved state-of-the-art results. Following [50], Wang *et al.* [59] introduced contrastive learning into this bilateral-branch network, which further improved the long-tailed classification performance.

5.2.3 Other Methods

Many other types of methods like meta-learning [138], [139], transfer learning [140] and knowledge distillation [141]. Some researchers are inspired by these methods and have proposed a series of approaches. For example, Wang *et al.* [101] utilize the head classes to train the meta-learner and transfer the learned knowledge to tail classes. Liu *et al.* [47] and Yin *et al.* [51] transfer the intra-class variance from head to tail to augment the diversity of tail classes. Most recently, Li *et al.* [142] use the supervised and self-supervised information to train a model to generate soft labels for all the data, and then distill a new student model with the generated soft labels and the original hard labels. The aforementioned

methods have achieved satisfying results, yet, they increase model complexity and/or the optimization difficulties.

5.3 Proposed Approach: GCL

The key idea of our proposed GCL is to utilize the softmax saturation to automatically balance the valid samples of head and tail classes. The theoretical motivation and the formulation of the loss function of the proposed approach are presented as follows.

5.3.1 Motivation

Figure 5.1 shows that the obscure region among different classes, especially the tail class, is large. One important factor of this obscure region is the softmax saturation in CE loss [124]. Suppose $\{x, y\} \in \mathcal{T}$ represents a sample $\{x, y\}$ from the training set $\mathcal{T}$ with the total N samples in C classes, and $y \in \{1, \ldots, C\}$ is the ground truth label. The softmax loss function for the input image x can be written as:

$$\mathcal{L}(x) = -\log p_y, \text{ with } p_y = \frac{e^{z_y}}{\sum_{j=1}^{C} e^{z_j}}, \tag{5.1}$$

where z_j represents the predicted logit of class j. We use the subscript y $(j \neq y)$ to represent the target class. That is, z_y indicates the target logit and $z_j (j \neq y)$ is the non-target logit.

In backward propagation, the gradients on z_j is calculated by:

$$\frac{\partial \mathcal{L}}{\partial z_j} = \begin{cases} p_j - 1, & j = y \\ p_j, & j \neq y. \end{cases} \tag{5.2}$$

Without loss of generality, we use the binary classification as an example. Supposing x is from class 1, the gradients on z_1 is then calculated by:

$$\frac{\partial \mathcal{L}}{\partial z_1} = -\frac{1}{1 + e^{z_1 - z_2}}. \tag{5.3}$$

Equation (5.3) indicates that the gradient of the target class rapidly approaches zero with the increase of the logit difference. Softmax can only slightly separate various classes,

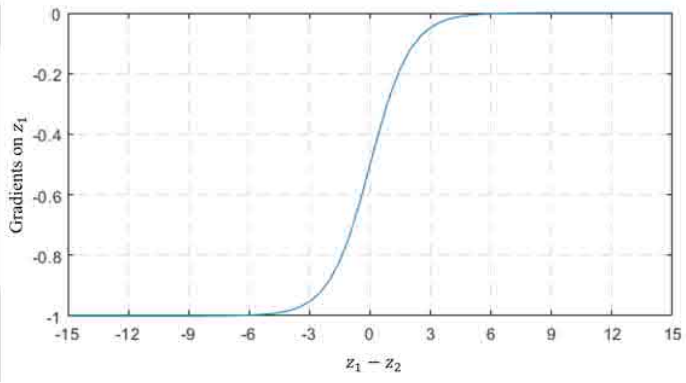


Figure 5.3: The gradient on z_y $(y = 1)$ in binary classification case. As the logit difference increases, the gradient rapidly approaches zero.

and lacks the power to evenly distribute each class in the embedded space. Therefore, there are many overlapping areas among the classes. In particular, under the circumstances of long-tailed classification, the tail class features are not sufficient to cover the real distribution in embedding space. The early gradient vanish caused by softmax saturation exacerbates the squeezing of their embedding space. A straightforward approach is to introduce hard margin [67], [95], [125]. However, the hard margin will cause the samples to shrink towards the class anchor and easy to overfit tail classes, which cannot evenly distribute the embedding space well. Fortunately, softmax saturation can help filter out the head class samples and make the tail class samples fully participate in training. In this way, the tail classes can be pushed away from the head classes and indirectly enlarge their embedding space.

5.3.2 Embedding Space Calibration

Suppose the features of different class samples satisfy Gaussian distribution. We can obtain a disturbed feature $\mathbf{f}^{cld}$ of the input by Gaussian sampling, which is represented as:

$$\mathbf{f}^{cld} \triangleq \mathbf{f} + \delta\mathbf{E}, \tag{5.4}$$

where $\mathbf{f} \in \mathbb{R}^D$ is the feature obtained from the embedding layer with the dimension of D. $\mathbf{E} \sim \mathcal{N}(\mathbf{u}, \mathbf{\Sigma})$ is the disturbance sampled from Gaussian distribution, and the mean vector

and covariance matrix are represented by $\mathbf{u} \in \mathbb{R}^D$ and $\mathbf{\Sigma} \in \mathbb{R}^{D\times D}$, respectively. $\delta > 0$ is a parameter that is used to adjust the amplitude of disturbance. In addition, δ should be a small number because a large disturbance will mislead the model. This disturbed feature is the input of the classifier. We use $\mathbf{W} = \{\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_C\} \in \mathbb{R}^{D\times C}$ to represent the weight matrix of the classifier, where $\mathbf{w}_j$ represents the anchor vector of class j in the classifier. Then, the corresponding disturbed logit z_j^{cld} of class j is calculated by:

$$\begin{aligned} z_j^{cld} &= \mathbf{w}_j^T \mathbf{f}^{cld} + \mathbf{b}_j \\ &= \mathbf{w}_j^T \mathbf{f} + \mathbf{b}_j + \mathbf{w}_j^T(\delta \mathbf{E}) \\ &= z_j + \delta(\mathbf{w}_j^T \mathbf{E}). \end{aligned} \tag{5.5}$$

As the range of z_j^{cld} is enlarged with random Gaussian disturbances, we call it Gaussian clouded logit, and $\delta(\mathbf{w}_j^T \mathbf{E})$ is the clouded term. Please note that the clouded term has the different degrees of influence on the final predicted results based on different predicted logits. It has a relatively small impact on z_j^{cld} when the original logit z_j is large. On the contrary, it will play a key role for z_j^{cld} when z_j is small. As a result, we need to normalize the effect caused by different predicted logits and maintain the consistency of the influence of the clouded term. Inspired by [5], [95], [98], we normalize the clouded logits based on cosine distance. In this way, the norm of the feature and the class anchor can be normalized to the fixed numbers. We use s_1 and s_2 to represent these two numbers. The normalized clouded logit is named *clouded cosine logit*, which is calculated by:

$$\begin{aligned} \tilde{z}_j^{cld} &= \frac{s_1 \mathbf{w}_j^T \cdot s_2 \mathbf{f}^{cld}}{\|\mathbf{w}_j^T\| \|\mathbf{f}^{cld}\|} \\ &= s \cdot \left(\frac{\mathbf{w}_j^T \mathbf{f}}{\|\mathbf{w}_j^T\| \|\mathbf{f} + \delta \mathbf{E}\|} + \delta \frac{\mathbf{w}_j^T \mathbf{E}}{\|\mathbf{w}_j^T\| \|\mathbf{f} + \delta \mathbf{E}\|}\right) \end{aligned}, \tag{5.6}$$

where $s = s_1 \cdot s_2$ is a constant. In the first term of Equation (5.6), $\|\mathbf{f} + \delta\mathbf{E}\| \approx \|\mathbf{f}\|$ because δ is a small number. In the second term, the norm of feature is normalized to s_1. Thus, $\tilde{z}_j^{cld}$

can be simplified as:

$$\tilde{z}_j^{cld} \approx s \cdot (\frac{\mathbf{w}_j^T \mathbf{f}}{\|\mathbf{w}_j^T\| \|\mathbf{f}\|} + \frac{\delta}{s_1} I_j \mathbf{E}), \tag{5.7}$$

where I_j is the identity vector that has the same direction as $\mathbf{w}_j^T$. In order to simplify the calculation, we make the clouded cosine logit still satisfy the Gaussian distribution. Thus, we introduce a constant σ and set the covariance matrix $\mathbf{\Sigma} = \sigma \mathbf{I}$, where $\mathbf{I} \in \mathcal{R}^{D \times D}$ is the identity matrix. Then, $I_j \mathbf{E}$ is the projection of the noise sampled by Gaussian in the direction of the anchor vector of class j. We denote its magnitude by ε_j. Therefore, $\tilde{z}_j^{cld}$ can be calculated by:

$$\begin{aligned} \tilde{z}_j^{cld} &= s \cdot (\tilde{z}_j + \tfrac{\delta}{s_1} \varepsilon_j) \\ &\Leftrightarrow s \cdot (\tilde{z}_j + \delta_j \varepsilon) \end{aligned}, \tag{5.8}$$

where $\tilde{z}_j = \cos \theta_j$ is the cosine distance, and θ_j is the angle between $\mathbf{f}$ and $\mathbf{w}_j$. δ_j is the logit cloud size that depends on different classes.

To achieve the two goals mentioned in Section 5.3.1, *i.e.*, 1) encourage tail class samples to participate more in training; 2) enlarge the embedding space for the tail classes, the size of logit cloud should be negatively correlated with the number of training samples. For the most frequent class, the diversity of training samples is sufficient and we set its logit cloud size to zero, while utilizing larger cloud sizes for tail classes. The merits of this large relative cloud size of tail classes are three-fold: 1) reduce the softmax saturation and thereby increase the training degree of tail classes; 2) different values are sampled randomly from the Gaussian cloud so as to avoid overfitting; 3) enlarge the margin of class boundary for tail classes and can calibrate the distortion of the embedding space. We therefore empirically set the cloud size for class j as:

$$\delta_j = \log n_{max} - \log n_j, \tag{5.9}$$

where n_{max} is the sample numbers of the most frequent class. We experimentally verify the effectiveness of this cloud size adjustment strategy in Section 5.4.4.2 .

The Gaussian clouded logit difference Δ_{y_j} between the target and non-target classes is:

$$\begin{aligned}\Delta_{y\_j} &= z_y^{cld} - z_j^{cld} \\ &= z_y - z_j + \varepsilon(\delta_y - \delta_j)\end{aligned}. \tag{5.10}$$

If $\varepsilon > 0$, Δ_{y_j} for tail classes will be increased. However, our goal is to reduce the logit difference to alleviate the softmax saturation for tail classes. In addition, a reduced logit corresponds to the feature that is relatively far from the class anchor. If the relatively distant feature can be predicted correctly, the closer one will be able to assign the right label. Therefore, we require ε to be negative. Subsequently, the clouded cosine logit can be written in the following form:

$$\tilde{z}_j^{cld} = s \cdot (\tilde{z}_j - \delta_j \|\varepsilon\|). \tag{5.11}$$

Taking the clouded cosine logit into the original softmax, we can obtain the loss function of GCL:

$$L_{GCL} = -\frac{1}{N} \sum_i \log \frac{e^{\tilde{z}_{y_i}^{cld}}}{\sum_j e^{\tilde{z}_j^{cld}}}. \tag{5.12}$$

5.3.3 Classifier Re-balance

The gradients derived in Equation (5.2) demonstrate that the sample of the target class y punishes the classifier weights $\mathbf{w}_j$ of non-target class $j, j \neq y$ w.r.t. p_j. The head classes have enormously greater training instances than tail classes. Therefore, the classifier weights of tail classes receive much more penalty than positive signals during training. Consequently, the classifier will bias towards the head classes, and the predicted logits of the tail classes will be seriously suppressed, resulting in low classification accuracy of the tail classes. A straightforward approach is to use the re-sampled data to re-train the classifier. We apply the classifier re-training (cRT), which was adopted by Kang *et al.* [12] and Wang *et al.* [55]. As the GCL loss enables different class samples to participate in training to different degrees, the effectiveness of different class samples is varied. Class-balanced sampling will lead to repeat training for tail classes. Drawing on the effective number

Algorithm 5.1: GCL with CBEN

Input: Training dataset $\mathcal{T}$;
Output: Predicted labels;

Initialize the model parameters ω of the CNN network $\phi((x, y); \omega)$ randomly ;
for $iter = 1$ *to* I_0 **do**
  Sample a batch samples $\mathcal{B}$ from the original long-tailed data $\mathcal{T}$ with batch size b;
  Calculate the logit cloud size δ_j by Equation (5.9): $\delta_j \leftarrow \log n_{max} - \log n_j$;
  Calculate the loss by Equation (5.12): $\mathcal{L}((x, y); \omega) = \frac{1}{b} \sum_{(x,y)\in\mathcal{B}} L_{GCL}(x, y)$;
  Update model parameters: $\omega = \omega - \alpha \nabla_\omega \mathcal{L}((x, y); \omega)$.
end
for $iter = I_0 + 1$ *to* $I_0 + I_1$ **do**
  Calculate sampling rate by Equation (5.15), Equation (5.13) and Equation (5.14): $\beta_j \leftarrow b \times \frac{\delta_j - \delta_{max}}{\delta_{max} - \delta_{min}} + a$; $\gamma_j \leftarrow \frac{1 - \beta_j}{1 - \beta_j^{n_j}}$; $\gamma_j \leftarrow \frac{\gamma_j}{\sum_i \gamma_i}$;
  Sample a batch samples $\mathcal{B}'$ with the sampling probability γ_j and the batch size b;
  Calculate the loss by Equation (5.12): $\mathcal{L}((x, y); \omega) = \frac{1}{b} \sum_{(x,y)\in\mathcal{B}'} L_{GCL}(x, y)$;
  Update classifier parameters ω_{cls} (representation parameters are frozen): $\omega_{cls} = \omega_{cls} - \alpha \nabla_{\omega_{cls}} \mathcal{L}((x, y); \omega_{cls})$.
end

proposed by Cui *et al.* [35], we propose the class-based effective number (CBEN) sampling to avoid excessive training of tail classes. The sampling probability γ_j of a sample from class j is calculated by:

$$\gamma_j = \frac{1 - \beta_j}{1 - \beta_j^{n_j}}. \tag{5.13}$$

Since the sum of the sampling probability for all data needs to be 1, we normalize γ_j by

$$\gamma_j \leftarrow \frac{\gamma_j}{\sum_i \gamma_i}. \tag{5.14}$$

β_j reflects the validity of different class samples. The class samples with large cloud size participate more in training. Therefore, β_j is positively correlated with cloud size δ_j. We set β_j as:

$$\beta_j = b \times \frac{\delta_j - \delta_{min}}{\delta_{max} - \delta_{min}} + a, \tag{5.15}$$

so that β_j can be in the region $[a, a + b]$, where a and b are the range hyper-parameters.

The overall training procedure of the proposed method is summarized in Algorithm 5.1.

5.4 Experiments

This section firstly gives a brief introduction to the five long-tailed datasets used in our experiments. Then, some key implementation details of the experiments are presented. After that, the classification accuracy of our proposed GCL and state-of-the-art methods are compared. Finally, the ablation studies are conducted to illustrate the properties of our proposed method.

5.4.1 Experimental Setting

The pre-setting parameters in the first stage were the Gaussian distribution parameters (μ, σ^2) and the region $[a, b]$ of sample validity β_j. We know that $\tilde{z}_i \in [-1, 1]$, thus the maximum feature cloud size cannot exceed 1. Since Gaussian distribution has a probability of about 99.7% falling in $[\mu - 3\sigma, \mu + 3\sigma]$, we set $\mu = 0$ and $\sigma = \frac{1}{3}$. We further clamped the ε to $[-1, 1]$ to prevent its amplitude from exceeding 1. We set $\beta_j \in [0.999, 0.9999]$, *i.e.*, $a = 0.999$ and $b = 0.0009$. Moreover, we normalized $\delta_i, i = \{1, 2, \cdots, C\}$ by $\delta_i \triangleq \delta_i / \delta_{max}$ to ensure that the maximum value of δ_i did not exceed 1. Similar with Zhong *et al.* [44], the mixup [42] strategy was also adopted in our experiments.

We utilized PyTorch [112] to implement all the backbones. SGD optimizer with momentum of 0.9 and the multi-step learning rate schedule were adopted. All the models were trained from scratch except ResNet-152 that was pre-trained on the original balanced version of ImageNet-2012. For the first stage, we selected ResNet-32 as the backbone network and followed the setting in Cao *et al.* [67] for CIFAR-10/100-LT. For the large-scale dataset, namely ImageNet-LT, iNaturalist 2018, and Places-LT, we mainly followed Kang *et al.* [12] except the learning rate schedule. For the second stage, *i.e.*, re-balancing the classifier, we followed Kang *et al.* [12] for all datasets.

The training details of GCL are summarized in Table 5.1.

Table 5.1: Summary of Basic Setting for GCL

Dataset	CIFAR-10/100-LT	ImageNet-LT	iNaturalist 2018	Places-LT
Obtaining Representation				
Backbone	ResNet-32	ResNet-50	ResNet-50	ResNet-152
Bini-bactch size	128	256	256	512
Initial lr	0.1	0.1	0.1	0.2
Weight decay	2×10^{-4}	2×10^{-4}	1×10^{-4}	5×10^{-4}
Training epochs	200	200	180	90
lr rate schedule	multi-step	multi-step	multi-step	multi-step
Decay ratio of lr	0.01	0.01	0.1	0.1
Decay epochs of lr	{160,180}	{160,180}	{120,160}	{60,80}
Re-balancing Classifier				
Training epochs	10	10	30	30
lr rate schedule	cosine	cosine	cosine	cosine
Decay ratio of lr	0.2	0.1	0.1	0.1

5.4.2 Competing Methods

To verify the effectiveness of the proposed method, we have conducted extensive experiments to compare with the previous methods, including the following two groups:

- **Baseline Methods**. We implemented vanilla training with cross-entropy (CE) loss as one of our baseline methods. Many visual recognition works [43], [48], [143], [144] have shown the efficacy of mixup, CE loss cooperated with mixup was therefore also compared.
- **State-of-the-art Methods**. The recently proposed representation learning method, namely OLTR [82] and logit adjustment method, namely De-confound-TDE inference [77] were compared. We also compared with the two-stage methods including LDAM-DRW [67] and MisLAS [44], which both achieve satisfactory classification accuracy on the aforementioned long-tailed datasets. For CIFAR-10/100-LT datasets, we made comparison with BBN [50] and contrastive learning [59]. For the large-scale datasets, we compared with the most recently proposed two-stage methods, including decoupling [12], logit adjustment [76] and DisAlign [56]. For a fair comparison, we additionally conducted the comparison experiment with the two-stage strategy which added classifier re-training (cRT) [12] to CE loss + mixup on all datasets.

Table 5.2: Comparison Results on CIFAR-10/100-LT.

Dataset	CIFAR-10-LT			CIFAR-100-LT		
Backbone Net	ResNet-32					
ρ	200	100	50	200	100	50
CE loss (baseline)	65.68	70.70	74.81	34.84	38.43	43.9
CE loss + mixup [42] (2018)	65.84	72.96	79.48	35.84	40.01	45.16
LDAM-DRW [67] (2019)	73.52	77.03	81.03	38.91	42.04	47.62
De-confound-TDE* [77] (2020)	-	80.60	83.60	-	44.15	50.31
CE loss + mixup + cRT [12] (2020)	73.06	79.15	84.21	41.73	45.12	50.86
BBN [50] (2020)	73.47	79.82	81.18	37.21	42.56	47.02
Contrastive learning* [59] (2021)	-	81.40	**85.36**	-	46.72	51.87
MisLAS [44] (2021)	**77.31**	**82.06**	85.16	**42.33**	**47.50**	**52.62**
GCL (ours)	**79.03**	**82.68**	**85.46**	**44.88**	**48.71**	**53.55**

Note: Top-1 accuracy (%) is reported. The best and the second-best results are shown in **underline bold** and **bold**, respectively.
* The results are quoted from the corresponding references. The other results are obtained by re-implementing with the official codes.

5.4.3 Comparison Results

Comparative studies have been conducted to show the efficacy of the proposed GCL. The results are presented in Table 5.2 and Table 5.3. We use top-1 accuracy on test sets as the performance metric. For the results from those papers that have yet to release the code or relevant hyper-parameters, we directly quote their results from the original papers.

5.4.3.1 Experimental Results on Small-scale Datasets

The results on CIFAR-10/100-LT datasets are summarized in Table 5.2. We can observe that our proposed GCL outperforms the previous methods by notable margins with all imbalanced ratios. Especially for the largest one, *i.e.*, $\rho = 200$, the proposed approach has obvious improvement. We get 79.03% and 44.88% in top-1 classification accuracy for CIFAR-10-LT and CIFAR-100-LT with $\rho = 200$, which surpasses the second best method, *i.e.*, MisLAS by a significant margin of 1.72% and 2.55%, respectively.

5.4.3.2 Experimental Results on Large-scale Datasets

The results on three large-scale long-tailed datasets, *i.e.*, ImageNet-LT, iNaturalist 2018, and Places-LT, are reported in Table 5.3. Our approach is superior to prior art on all datasets. On ImageNet-LT, our method achieves 54.88% top-1 accuracy, outperforming DisAlign

Table 5.3: Comparison Results on ImageNet-LT, iNaturalist 2018 and Places-LT.

Dataset	ImageNet-LT	iNaturalist 2018	Places-LT
Backbone Net	ResNet-50	ResNet-50	ResNet-152
CE loss (baseline)	44.51	63.80	27.13
CE loss + mixup [42] (2018)	45.66	65.77	29.51
LDAM-DRW*[67] (2019)	48.80	68.00	-
OLTR* [82] (2019)	-	-	35.9
Decoupling [12] (2020)	47.70	69.49	37.62
CE loss + mixup + cRT [12] (2020)	51.68	70.16	38.51
Logit adjustment*[76] (2021)	51.11	66.36	-
DisAlign*[56] (2021)	**52.91**	70.06	39.30
MisLAS [44] (2021)	52.11	**71.57**	**40.15**
GCL (ours)	**54.88**	**72.01**	**40.64**

Note: Top-1 accuracy (%) is reported. The best and the second-best results are shown in **underline bold** and **bold**, respectively.
* The results are quoted from the corresponding references. The other results are obtained by re-implementing with the official codes.

by a large margin at 1.97% and MisLAS at 2.77%, respectively. On iNaturalist 2018, the proposed approach achieves 72.01% top-1 accuracy, which outperforms the second-best method by 0.44%. On Place-LT, our method achieves 40.64% top-1 classification accuracy, with a performance gain at 0.49% over MisLAS. Although the performance gain compared with MisLAS on iNaturalist 2018 and Place-LT is not as high as other datasets, our method does not require hyper-parameters searching for different datasets, and thus it is relatively easy to implement.

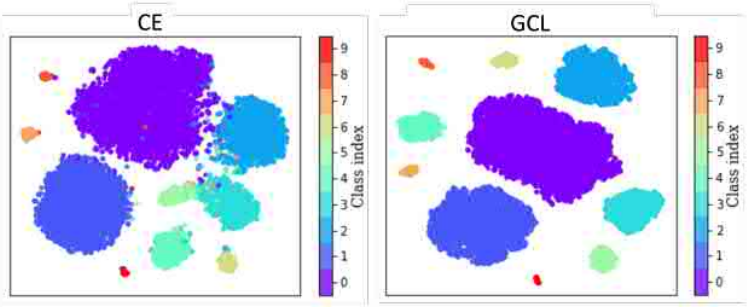
5.4.4 Model Validation and Analysis

We conduct a series of ablation studies to further analyze the proposed method.

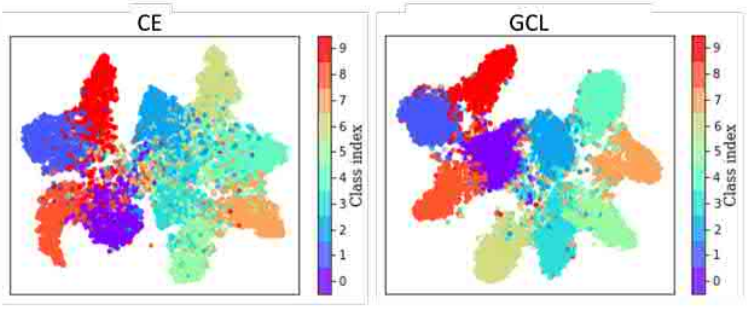
5.4.4.1 The Role of Gaussian Clouded Logit

In order to obtain additional insight, we utilized t-SNE projection of the embedding for visualization. Since the loss functions of baseline and MisLAS are both CE loss and MisLAS performed the second-best in most cases we have tried so far, we visualized CE loss embedding for comparison. The embeddings were calculated from the samples in CIFAR-10-LT with $\rho = 100$. Figure 5.4 shows the visualization results on the training and test set. From the result of the training set (Figure 5.4a), we can see that the embeddings

obtained via GCL of different classes are more scattered. Therefore, the GCL embedding of each class is much easier to separate. The results of the test set shown in Figure 5.4b justify the efficacy of our proposed approach. The obscure region of CE loss embedding is larger than that of GCL embedding. Good embedding helps improve the model performance. We only re-fine the classifier with the simple cRT without any other complicated technologies, but the classification accuracy can be improved a lot.



(a) On training set



(b) On test set

Figure 5.4: Visualization of the embedding via t-SNE from CIFAR-10-LT with $\rho = 100$, where backbone network is ResNet-32. (Color for the best view.)

5.4.4.2 Cloud Size Adjustment Strategy

We explored several different cloud size adjustment strategies, which included cosine form, power difference with different exponents (exp.:1/3 and exp.:1/4), and logarithmic difference. For a fair comparison, the sampler and re-training strategy were selected as CBEN and cRT, respectively. Table 5.4 shows the results. We choose the log. diff. strategy according to Table 5.4.

Table 5.4: Ablation Experiment of Different Cloud Size Adjustment Strategies.

Adjustment strategies	Expression	Accuracy(%)
Cosine	$\cos(n_j/n_{max} \cdot \pi/2)$	79.21
Power difference (exp.:1/3)	$n_{max}^{1/3} - n_j^{1/3}$	80.80
Power difference (exp.:1/4)	$n_{max}^{1/4} - n_j^{1/4}$	82.31
Logarithm difference	$\log n_{max} - \log n_j$	**82.68**

Note: The experiments are conducted on CIFAR-10-LT with $\rho = 100$.

Table 5.5: Ablation Experiment of Different Re-sampling Strategies.

Re-sampling strategy	Re-training strategy	Accuracy(%)
Instance balance	cRT	80.41
Class balance	cRT	82.43
Effective number	cRT	82.47
CBEN	cRT	**82.68**

Note: The experiments are conducted on CIFAR-10-LT with $\rho = 100$.

5.4.4.3 Classifier Re-balance Strategies

We compared different strategies of data re-sampling and the classifier re-training to better analyze our proposed method. The re-sampling strategy included: instance balance [12], class balance [12], class balance with effective number [35], and our proposed class-based effective number (CBEN). For a fair comparison, the re-training strategies for all samplers were cRT. Table 5.5 shows the effectiveness of CBEN. For the selection of classifier re-training strategy, we first trained the backbone without any classifier re-training technology. Then, we fixed the representation and re-balance the classifier with learnable weight scaling (LWS) [12], τ-normalization [12], and cRT, respectively. Table 5.6 presents the top-1 accuracy of CIFAR-10-LT with $\rho = 100$. We can observe that, even without any classifier re-training technique, our approach can still beat most state-of-the-arts including two-stage methods. For example, our GCL without classifier re-training suppresses BBN by 0.7%. Further, cRT performs the best among the classifier re-training strategies, which improves the top-1 accuracy by 1.64%. From Table 5.5 and Table 5.6, we can observe that instance balance with cRT degrades model performance, which indicates that training the classifier

Table 5.6: Ablation Experiment of Different Re-training Strategies.

Re-sampling strategy	Re-training strategy	Accuracy(%)
-	w/o RT	80.52
CBEN	LWS	82.25
CBEN	τ-normalization	82.16
CBEN	cRT	**82.68**

Note: The experiments are conducted on CIFAR-10-LT with $\rho = 100$.

Table 5.7: Classification Accuracy on Different Scale Classes

Method	Head	Middle	Tail	All
CE loss (baseline)	**64.91**	38.10	11.28	44.51
CosFace [98] (2018)	64.48	39.26	11.55	44.94
ArcFace [95] (2019)	**64.86**	38.07	11.75	44.54
LDAM-DRW [67] (2019)	58.63	48.95	30.37	49.96
OLTR [82] (2019)	61.93	44.68	19.98	47.72
Decoupling [12] (2020)	63.71	43.01	20.55	47.70
LADE [115] (2021)	62.96	47.72	32.75	51.40
MisLAS [44] (2021)	62.43	**49.31**	**33.89**	**52.11**
GCL (ours)	63.78	**52.62**	**38.70**	**54.88**

Note: Top-1 classification accuracy (%) is reported. The experiments are conducted on ImageNet-LT. The best and the second-best results are shown in **underline bold** and **bold**, respectively.
Head class has more than 100 training images; *middle class* has 20–100 training images; and *tail class* has less than 20 training images.

with instance balance may lead to classifier overfitting.

5.4.4.4 Classification Accuracy on Different Scale Classes

We have conducted the experiments on ImageNet-LT to obtain the accuracy towards different scale classes. The results are shown in Table 5.7, where we present the results of re-margining methods and the latest class re-balancing methods.

The results of CosFace [98] and ArcFace [95] that are for balanced data drop a lot on middle and tail classes. LDAM-DRW [67] and LADE [115] increase the accuracy of medium and few shot classes, but decreases that of head classes a lot. GCL outperforms the state-of-the-arts on middle and tail classes with large margins. Meanwhile, the accuracy of the head class decreases the least. Significantly improving the accuracy of tail classes while preventing that of the head classes from diminishing illustrates the superiority of our approach.

5.5 Concluding Remarks

In this chapter, we have proposed the novel GCL as well as simple but effective CBEN classifier re-training strategy to further improve the classification accuracy on long-tailed distribution data.

We have found that softmax saturation reduces sample validity, which has different effects on head and tail classes. This implies that, from another perspective, softmax saturation can be utilized to automatically adjust the training sample validity of different classes. Subsequently, we have proposed the GCL. The tail class logits are set to relatively large cloud sizes to encourage more tail class samples to participate in training as well as leave large margins, which help obtain evenly distributed embedding space. The effectiveness of different classes is varied via GCL. Then, the simple but effective CBEN sampling strategy incorporated with cRT for classifier balancing has been proposed, which can further boost the model performance. Extensive experiments on various benchmark datasets have demonstrated that the proposed GCL has superior performance compared to the existing state-of-the-art methods.

The proposed GCL with CBEN still needs two separated stages to obtain further improvement. In our future work, we try to combine these two stages and propose a model that can be trained end-to-end while maintaining model performance.

Chapter 6

Conclusions and Future Work

This chapter concludes the thesis and discusses some potential research directions for the future.

6.1 Conclusions

Real-world data tends to have a long-tailed distribution. However, CNNs were originally designed on the assumption that the data is balanced, thus they can perform very well on large-scale annotated and pre-balanced datasets. Their performance drops dramatically on long-tailed data, which hinders the practical application. Therefore, the study of long-tailed data has both theoretical and practical significance because of the prevalent data imbalance in daily life. The most direct ways, such as re-sampling the data or re-weight the loss cannot handle the long-tailed problem well. Roughly increasing model complexity will lead to optimization difficulties, making them impractical. We need to seek deep theoretical studies for long-tailed data. The key issues of long-tailed learning are the over-suppressed embedding space for tail classes and the biased classifier towards the head classes. This thesis has proposed three methods from the perspective of loss function to tackle the aforementioned two key issues.

The first method, *i.e.*, KPS loss has studied the influence of different kinds of points in feature space towards CNNs for long-tailed classification. We have observed that key points are more important for classification. The proposed KPS loss with GA strategy can assign large margins for the key points and tail class samples. Meanwhile, GA can

re-balance the gradients of positive and negative samples for each class. As a result, KPS loss with GA achieves significant performance gains on tail classes with little drop in head classes.

The second method, *i.e.*, FBL has investigated the influence of feature norm towards CNNs trained on long-tailed dataset. The proposed FBL with curriculum learning which adds an extra class-based stimulus to the logit encourages larger feature norm for tail classes, thereby improving the generalization performance of these classes. Moreover, the stimulus intensity is gradually increased. This robust training strategy not only enhances the classification accuracy of tail classes to a large extent, but also maintains the performance of head classes. The observations towards feature norm not only justify our motivation about the influence of feature norm on decision margin, but also offer a new way to investigate long-tailed learning.

The third method, *i.e.*, GCL has studied the effect of softmax saturation on the model trained by long-tailed data. GCL perturbs different class logits with varied amplitude to automatically adjust the training sample validity of each class. To alleviate the bias in a classifier, the simple but effective CBEN sampling strategy incorporated with cRT for classifier balancing has been proposed. GCL with CBEN has superior performance compared to the existing state-of-the-art methods.

6.2 Future Directions

The studies conducted in this thesis mainly exploit the loss function modification and logit adjustment methods. This kind of methods are simple but effective, and does not increase the model complexity. However, as discussed in Chapter 1 and Figure 1.2 shown, loss function is only the last step of the model training. And existing solutions mainly pay attention to one or two key parts in the whole training process and do not consider model training as a whole. Additionally, the measurements for the performance of the methods are inappropriate. The classification performance cannot be accurately evaluated using traditional assessment metrics such as accuracy or error rate, which cannot reflect the per-class performance. The per-class accuracy or error rate is too verbose, especially

for large-scale datasets with thousands of classes. And class sizes vary widely across datasets. The head, middle and tail classes can only reflect the relative size of the class, but cannot clearly reflect the number samples in each class. For example, the head class in CIFAR-100-LT with $\rho = 100$ has 500 samples, which is the number of middle class in CIFAR-10-LT with $\rho = 100$. Furthermore, most existing methods focus on clean data that has clean label. The noisy and contaminate data are more prevalent in real-world. Therefore, we would like to conduct research on the above issues in our future work. The following highlights several potential directions for future studies.

High performance algorithm

Considering the entire training process as a whole can help improve model performance. Our KPS and FBL take the gradient and loss into consideration, but without considering the sampling strategy and model. GCL re-design the sampling strategy. Nevertheless, it utilizes the two-stage strategy that cannot be trained end-to-end. Many other methods like decoupling representation based methods [12], [56], LDAM [67] and meta balanced softmax [31], *etc*.introduced in Chapter 1 combine several types of strategies. They focus on several parts but still have no overall re-design of the training process. We can consider long-tail visual recognition from the sampling, model, loss function and optimization process as a whole to obtain a model that can be trained end-to-end and improve the performance in all classes. Besides the methods introduced in Section 1.2, other methods, for example, feature selection algorithm [145], [146], saliency detection[147], [148] and visual transformer [149], [150] can also be considered.

Concise and unified metrics

Since the accuracy/error rate cannot properly access the model performance for long-tailed learning, other evaluation metric can be considered. For example, in binary classification, F1 score, G-mean, Receiver Operating Characteristics (ROC) [151] and area under the curve (AUC) are utilized as the supplementary evaluation metrics for accuracy/error rate. We can extend these metrics to multi-classes. In addition, Zhong *et al.* [44] suggested the Expected calibration error (ECE) which measures the calibration of

network. It can also effectively reflect the performance of the model on long-tail data in terms of data attributes. Moreover, it is meaningful to take these metrics as the optimization objectives when re-designing the imbalance-aware models.

Long-tailed noisy data

For supervised learning, how to obtain a large amount of labeled data is undoubtedly a key issue. The manual labeling method is time-consuming and labor-intensive, while some automated methods (such as directly using labeled pictures on social networks) can be quickly obtained. There are a large number of samples, but the labels cannot be guaranteed to be accurate, and they may be contaminated or noisy. Direct feeding such dataset into CNNs will make the model overfit to the noisy samples, resulting in poor generalization performance. A lot of work (for example, see [152]–[154]) has investigated this issue. However, to the best of our knowledge, few works have studied the noisy label under the long-tailed data. The noisy long-tailed data is a more challenging task and worth studying. The effect of noise in the data on the long-tailed learning is unclear. Some work [100], [124], [155] has shown the beneficial aspects of noise for model training. How to exploit the noise in noisy long-tail data and methods, such as self-supervision [156], [157] and contrastive learning [158], [159], are worth studying.

Bibliography

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2016, pp. 770–778.

[2] X. Liu and Y.-m. Cheung, "Learning multi-boosted hmms for lip-password based speaker verification," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 2, pp. 233–246, 2014.

[3] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Venice,Italy: IEEE Computer Society, Oct. 2017, pp. 2961–2969.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2016.

[5] F. Wang, X. Xiang, J. Cheng, and A. L. Yuille, "Normface: L_2 hypersphere embedding for face verification," in *Proceedings of ACM International Conference on Multimedia (MM)*, ser. MM'17, Mountain View, California, USA: ACM, Dec. 2017, pp. 1041–1049.

[6] J. Gu, Z. Wang, J. Kuen, *et al.*, "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354–377, 2018.

[7] S. Xie, R. B. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of IEEE Conference on Computer*

Vision and Pattern Recognition (CVPR), Honolulu, HI, USA: Computer Vision Foundation / IEEE Computer Society, 2017, pp. 5987–5995.

[8] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[9] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 6, pp. 1452–1464, 2017.

[10] M. G. Kendall *et al.*, *The advanced theory of statistics*. Charles Griffin and Co., Ltd., 42 Drury Lane, London, 1948.

[11] Y. Zhang, B. Kang, B. Hooi, S. Yan, and J. Feng, "Deep long-tailed learning: A survey," *arXiv preprint arXiv:2110.04596*, 2021.

[12] B. Kang, S. Xie, M. Rohrbach, Z. Yan, A. Gordo, J. Feng, and Y. Kalantidis, "Decoupling representation and classifier for long-tailed recognition," in *Proceedings of International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia: OpenReview.net, Apr. 2020.

[13] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[14] A. Estabrooks, T. Jo, and N. Japkowicz, "A multiple resampling method for learning from imbalanced data sets," *Computational Intelligence*, vol. 20, no. 1, pp. 18–36, 2004.

[15] H. Han, W. Wang, and B. Mao, "Borderline-smote: A new over-sampling method in imbalanced data sets learning," in *Advances in Intelligent Computing, International Conference on Intelligent Computing, ICIC*, ser. Lecture Notes in Computer Science, vol. 3644, Hefei, China: Springer, Aug. 2005, pp. 878–887.

[16] X. Liu, J. Wu, and Z. Zhou, "Exploratory undersampling for class-imbalance learning," *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics*, vol. 39, no. 2, pp. 539–550, 2009.

[17] Z. Zhang and T. Pfister, "Learning fast sample re-weighting without reward data," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada: IEEE Computer Society, Oct. 2021, pp. 705–714.

[18] J. Byrd and Z. Lipton, "What is the effect of importance weighting in deep learning?" In *Proceedings of International Conference on Machine Learning (ICML)*, ser. Proceedings of Machine Learning Research, vol. 97, Long Beach, California, USA: PMLR, Jun. 2019, pp. 872–881.

[19] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Networks*, vol. 106, pp. 249–259, 2018.

[20] H. He, Y. Bai, E. A. Garcia, and S. Li, "Adasyn: Adaptive synthetic sampling approach for imbalanced learning," in *Proceedings of the International Joint Conference on Neural Networks, IJCNN, part of the IEEE World Congress on Computational Intelligence, WCCI*, Hong Kong, China: IEEE Computer Society, Jun. 2008, pp. 1322–1328.

[21] N. Sarafianos, X. Xu, and I. A. Kakadiaris, "Deep imbalanced attribute classification using visual attention aggregation," in *Proceedings of European conference on computer vision (ECCV)*, ser. Lecture Notes in Computer Science, vol. 11215, Munich, Germany: Springer, Sep. 2018, pp. 708–725.

[22] I. Tomek, "Two modifications of cnn," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 6, pp. 769–772, 1976.

[23] M. Kubat, S. Matwin, *et al.*, "Addressing the curse of imbalanced training sets: One-sided selection," in *Proceedings of International Conference on Machine*

Learning (ICML), vol. 97, Nashville, Tennessee, USA: Morgan Kaufmann, Jul. 1997, pp. 179–186.

[24] J. Laurikkala, "Improving identification of difficult small classes by balancing class distribution," in *Artificial Intelligence Medicine, Conference on AI in Medicine in Europe, AIME*, ser. Lecture Notes in Computer Science, vol. 2101, Cascais, Portugal: Springer, Jul. 2001, pp. 63–66.

[25] G. E. A. P. A. Batista, A. L. C. Bazzan, and M. C. Monard, "Balancing training data for automated annotation of keywords: A case study," in *Brazilian Workshop on Bioinformatics (WOB)*, Macaé, RJ, Brazil, Nov. 2003, pp. 10–18.

[26] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *SIGKDD Explor.*, vol. 6, no. 1, pp. 20–29, 2004.

[27] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, Lake Tahoe, Nevada, United States: Curran Associates, Inc., Nov. 2013, pp. 3111–3119.

[28] D. Mahajan, R. B. Girshick, V. Ramanathan, K. He, M. Paluri, Y. Li, A. Bharambe, and L. v. d. Maaten, "Exploring the limits of weakly supervised pretraining," in *Proceedings of European conference on computer vision (ECCV)*, ser. Lecture Notes in Computer Science, vol. 11206, Munich, Germany: Springer, Sep. 2018, pp. 185–201.

[29] Y. Cui, Y. Song, C. Sun, A. Howard, and S. Belongie, "Large scale fine-grained categorization and domain-specific transfer learning," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2018, pp. 4109–4118.

[30] Y. Wang, W. Gan, J. Yang, W. Wu, and J. Yan, "Dynamic curriculum learning for imbalanced data classification," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Seoul, Korea (South): IEEE Computer Society, Oct. 2019.

[31] J. Ren, C. Yu, S. Sheng, X. Ma, H. Zhao, S. Yi, and H. Li, "Balanced meta-softmax for long-tailed visual recognition," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, Virtual Event: Curran Associates, Inc., 2020.

[32] Y. Zang, C. Huang, and C. C. Loy, "Fasa: Feature augmentation and sampling adaptation for long-tailed instance segmentation," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada: IEEE Computer Society, Oct. 2021, pp. 3437–3446.

[33] Z. Liu, W. Cao, Z. Gao, J. Bian, H. Chen, Y. Chang, and T. Liu, "Self-paced ensemble for highly imbalanced massive data classification," in *International Conference on Data Engineering, ICDE*, Dallas, TX, USA: IEEE Computer Society, Apr. 2020, pp. 841–852.

[34] B. Li, Y. Liu, and X. Wang, "Gradient harmonized single-stage detector," in *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, Honolulu, Hawaii, USA: AAAI Press, Feb. 2019, pp. 8577–8584. [Online]. Available: https://doi.org/10.1609/aaai.v33i01.33018577.

[35] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2019, pp. 9268–9277.

[36] T. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 318–327, 2020.

[37] S. Park, J. Lim, Y. Jeon, and J. Y. Choi, "Influence-balanced loss for imbalanced visual classification," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada: IEEE Computer Society, Oct. 2021, pp. 715–724.

[38] M. Ren, W. Zeng, B. Yang, and R. Urtasun, "Learning to reweight examples for robust deep learning," in *Proceedings of International Conference on Machine Learning (ICML)*, vol. 80, 2018, pp. 4331–4340.

[39] J. Shu, Q. Xie, L. Yi, Q. Zhao, S. Zhou, Z. Xu, and D. Meng, "Meta-weight-net: Learning an explicit mapping for sample weighting," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, Vancouver, BC, Canada: Curran Associates, Inc., Dec. 2019, pp. 1917–1928.

[40] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, IEEE Computer Society, 2015, pp. 1–9. [Online]. Available: https://doi.org/10.1109/CVPR.2015.7298594.

[41] P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, and K. He, "Accurate, large minibatch SGD: training imagenet in 1 hour," *CoRR*, vol. abs/1706.02677, 2017. [Online]. Available: http://arxiv.org/abs/1706.02677.

[42] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," in *Proceedings of International Conference on Learning Representations (ICLR)*, Vancouver, BC, Canada: OpenReview.net, Apr. 2018, pp. 1–13.

[43] Y. Zhang, X. Wei, B. Zhou, and J. Wu, "Bag of tricks for long-tailed visual recognition with deep convolutional neural networks," in *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, Virtual Event: AAAI Press, Feb. 2021,

pp. 3447–3455. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/16458.

[44] Z. Zhong, J. Cui, S. Liu, and J. Jia, "Improving calibration for long-tailed recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Virtual Event: Computer Vision Foundation / IEEE Computer Society, Jun. 2021, pp. 16 489–16 498.

[45] H. Chou, S. Chang, J. Pan, W. Wei, and D. Juan, "Remix: Rebalanced mixup," in *Proceedings of European conference on computer vision Workshops (ECCVW)*, ser. Lecture Notes in Computer Science, vol. 12540, Glasgow, UK: Springer, Aug. 2020, pp. 95–110.

[46] J. Wang, T. Lukasiewicz, X. Hu, J. Cai, and X. Zhenghua, "RSG: A simple but effective module for learning imbalanced datasets," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Virtual Event: Computer Vision Foundation / IEEE Computer Society, Jun. 2021, pp. 3784–3793.

[47] J. Kim, J. Jeong, and J. Shin, "M2m: Imbalanced classification via major-to-minor translation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 13 896–13 905.

[48] J. Kim, W. Choo, H. Jeong, and H. O. Song, "Co-mixup: Saliency guided joint mixup with supermodular diversity," in *Proceedings of International Conference on Learning Representations (ICLR)*, Virtual Event, Austria, May 2021.

[49] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, and Y. Bengio, "Manifold mixup: Better representations by interpolating hidden states," in *Proceedings of International Conference on Machine Learning (ICML)*, ser. Proceedings of Machine Learning Research, PMLR, vol. 97, Long Beach, California, USA, Jun. 2019, pp. 6438–6447.

[50] B. Zhou, Q. Cui, X.-S. Wei, and Z.-M. Chen, "Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition," in *Proceedings of*

IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2020, pp. 9719–9728.

[51] X. Yin, X. Yu, K. Sohn, X. Liu, and M. Chandraker, "Feature transfer learning for face recognition with under-represented data," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2019, pp. 5704–5713.

[52] J. Liu, Y. Sun, C. Han, Z. Dou, and W. Li, "Deep representation learning on long-tailed data: A learnable embedding augmentation perspective," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2020.

[53] P. Chu, X. Bian, S. Liu, and H. Ling, "Feature space augmentation for long-tailed data," in *Proceedings of European conference on computer vision (ECCV)*, vol. 12374, Springer, 2020, pp. 694–710.

[54] B. Zhou, A. Khosla, À. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2016, pp. 2921–2929. [Online]. Available: https://doi.org/10.1109/CVPR.2016.319.

[55] T. Wang, Y. Li, B. Kang, J. Li, J. Liew, S. Tang, S. Hoi, and J. Feng, "The devil is in classification: A simple framework for long-tail instance segmentation," in *Proceedings of European conference on computer vision (ECCV)*, ser. Lecture Notes in Computer Science, vol. 12359, Glasgow, UK: Springer, Aug. 2020, pp. 728–744.

[56] S. Zhang, Z. Li, S. Yan, X. He, and J. Sun, "Distribution alignment: A unified framework for long-tail visual recognition," in *Proceedings of IEEE Conference*

on Computer Vision and Pattern Recognition (CVPR), Virtual Event: Computer Vision Foundation / IEEE Computer Society, Jun. 2021, pp. 2361–2370.

[57] R. Müller, S. Kornblith, and G. E. Hinton, "When does label smoothing help?" In *Proceedings of Neural Information Processing Systems (NeurIPS)*, Vancouver, BC, Canada: Curran Associates, Inc., Nov. 2019, pp. 4696–4705.

[58] B. Kang, Y. Li, S. Xie, Z. Yuan, and J. Feng, "Exploring balanced feature spaces for representation learning," in *Proceedings of International Conference on Learning Representations (ICLR)*, Virtual Event, Austria: OpenReview.net, May 2021. [Online]. Available: https://openreview.net/forum?id=OqtLIabPTit.

[59] P. Wang, K. Han, X. Wei, L. Zhang, and L. Wang, "Contrastive learning based hybrid networks for long-tailed image classification," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Virtual Event: Computer Vision Foundation / IEEE Computer Society, Jun. 2021, pp. 943–952.

[60] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," in *Proceedings of International Conference on Machine Learning (ICML)*, ser. Proceedings of Machine Learning Research, vol. 119, Virtual Event: PMLR, Jul. 2020, pp. 1597–1607. [Online]. Available: http://proceedings.mlr.press/v119/chen20j.html.

[61] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, Virtual Event: Curran Associates, Inc., Nov. 2020.

[62] H. Guo and S. Wang, "Long-tailed multi-label visual recognition by collaborative training on uniform and re-balanced samplings," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, virtual Event: Computer Vision Foundation / IEEE Computer Society, Jun. 2021, pp. 15 089–15 098.

[63] Y. Li, T. Wang, B. Kang, S. Tang, C. Wang, J. Li, and J. Feng, "Overcoming classifier imbalance for long-tail object detection with balanced group softmax," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2020, pp. 10 988–10 997.

[64] L. Xiang, G. Ding, and J. Han, "Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification," in *Proceedings of European conference on computer vision (ECCV)*, Springer, 2020, pp. 247–263.

[65] J. Cai, Y. Wang, and J.-N. Hwang, "Ace: Ally complementary experts for solving long-tailed recognition in one-shot," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada: IEEE Computer Society, Dec. 2021, pp. 112–121.

[66] Y. Zhang, B. Hooi, L. Hong, and J. Feng, "Test-agnostic long-tailed recognition by test-time aggregating diverse experts with self-supervision," *arXiv preprint arXiv:2107.09249*, 2021.

[67] K. Cao, C. Wei, A. Gaidon, N. Arechiga, and T. Ma, "Learning imbalanced datasets with label-distribution-aware margin loss," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, Jaipur, India: Curran Associates, Inc., Jun. 2019, pp. 1565–1576.

[68] S. H. Khan, M. Hayat, S. W. Zamir, J. Shen, and L. Shao, "Striking the right balance with uncertainty," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2019, pp. 103–112.

[69] T. Wu, Z. Liu, Q. Huang, Y. Wang, and D. Lin, "Adversarial robustness under long-tailed distribution," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Virtual Event: Computer Vision Foundation / IEEE Computer Society, Jun. 2021, pp. 8659–8668.

[70] T. Hsieh, E. Robb, H. Chen, and J. Huang, "Droploss for long-tail instance segmentation," in *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, Virtual Event: AAAI Press, Feb. 2021, pp. 1549–1557. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/16246.

[71] J. Tan, C. Wang, B. Li, Q. Li, W. Ouyang, C. Yin, and J. Yan, "Equalization loss for long-tailed object recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2020, pp. 11 662–11 671.

[72] J. Tan, X. Lu, G. Zhang, C. Yin, and Q. Li, "Equalization loss v2: A new gradient balance approach for long-tailed object detection," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Virtual Event: Computer Vision Foundation / IEEE Computer Society, 2021, pp. 1685–1694.

[73] T.-I. Hsieh, E. Robb, H.-T. Chen, and J.-B. Huang, "Droploss for long-tail instance segmentation," in *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, vol. 35, Virtual Event: AAAI Press, 2021, pp. 1549–1557.

[74] J. Wang, W. Zhang, Y. Zang, Y. Cao, J. Pang, T. Gong, K. Chen, Z. Liu, C. C. Loy, and D. Lin, "Seesaw loss for long-tailed instance segmentation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Virtual Event: Computer Vision Foundation / IEEE Computer Society, 2021, pp. 9695–9704.

[75] T. Wang, Y. Zhu, C. Zhao, W. Zeng, J. Wang, and M. Tang, "Adaptive class suppression loss for long-tail object detection," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Virtual Event: Computer Vision Foundation / IEEE Computer Society, 2021, pp. 3103–3112.

[76] A. K. Menon, S. Jayasumana, A. S. Rawat, H. Jain, A. Veit, and S. Kumar, "Long-tail learning via logit adjustment," in *Proceedings of International Conference on Learning Representations (ICLR)*, Austria, May 2021.

[77] K. Tang, J. Huang, and H. Zhang, "Long-tailed classification by keeping the good and removing the bad momentum causal effect," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, Virtual Event: Curran Associates, Inc., Dec. 2020.

[78] M. Glymour, J. Pearl, and N. P. Jewell, *Causal inference in statistics: A primer*. John Wiley & Sons, 2016.

[79] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," ser. Proceedings of International Conference on Machine Learning (ICML), New York, NY, USA: Morgan Kaufmann, 2009.

[80] A. Krizhevsky, G. Hinton, *et al.*, "Learning multiple layers of features from tiny images," *Tech Report*, 2009.

[81] A. Torralba, R. Fergus, and W. T. Freeman, "80 million tiny images: A large data set for nonparametric object and scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 11, pp. 1958–1970, 2008.

[82] Z. Liu, Z. Miao, X. Zhan, J. Wang, B. Gong, and S. X. Yu, "Large-scale long-tailed recognition in an open world," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2019, pp. 2537–2546.

[83] G. Van Horn, O. Mac Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, and S. J. Belongie, "The inaturalist species classification and detection dataset," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2018, pp. 8769–8778.

[84] Visipedia, *Inaturalist 2018 competition*, 2018. [Online]. Available: https://github.com/visipedia/inat_comp/tree/master/2018.

[85] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[86] Q. Zhong, C. Li, Y. Zhang, H. Sun, S. Yang, D. Xie, and S. Pu, "Towards good practices for recognition & detection," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition workshops (CVPRW)*, vol. 1, Computer Vision Foundation / IEEE Computer Society, 2016.

[87] N. Japkowicz, "The class imbalance problem: Significance and strategies," in *Procceeding of the International Conference on Artificial Intelligence. (IC-AI)*, Citeseer, vol. 56, 2000.

[88] B. C. Wallace, K. Small, C. E. Brodley, and T. A. Trikalinos, "Class imbalance, redux," in *Proceedings of IEEE International Conference on Data Mining (ICDM)*, Vancouver, BC, Canada: IEEE Computer Society, Dec. 2011, pp. 754–763.

[89] S. H. Khan, M. Hayat, M. Bennamoun, F. A. Sohel, and R. Togneri, "Cost-sensitive learning of deep feature representations from imbalanced data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3573–3587, 2018.

[90] C. Huang, Y. Li, C. C. Loy, and X. Tang, "Learning deep representation for imbalanced classification," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2016, pp. 5375–5384.

[91] A. Iranmehr, H. Masnadi-Shirazi, and N. Vasconcelos, "Cost-sensitive support vector machines," *Neurocomputing*, vol. 343, pp. 50–64, 2019.

[92] C. Feng, Y. Zhong, and W. Huang, "Exploring classification equilibrium in long-tailed object detection," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada: IEEE Computer Society, Oct. 2021, pp. 3417–3426.

[93] Z. Deng, H. Liu, Y. Wang, C. Wang, Z. Yu, and X. Sun, "Pml: Progressive margin loss for long-tailed age classification," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Virtual Event: Computer Vision Foundation / IEEE Computer Society, Jun. 2021, pp. 10 503–10 512.

[94] F. Wang, J. Cheng, W. Liu, and H. Liu, "Additive margin softmax for face verification," *IEEE Signal Processing Letters*, vol. 25, no. 7, pp. 926–930, 2018.

[95] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2019, pp. 4690–4699.

[96] B. Chen, W. Deng, and H. Shen, "Virtual class enhanced discriminative embedding learning," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, ontréal, Canada: Curran Associates, Inc., Dec. 2018, pp. 1946–1956.

[97] W. Liu, Y. Wen, Z. Yu, and M. Yang, "Large-margin softmax loss for convolutional neural networks," in *Proceedings of International Conference on Machine Learning (ICML)*, vol. 48, New York, New York, USA: PMLR, Jun. 2016, pp. 507–516.

[98] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu, "Cosface: Large margin cosine loss for deep face recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA: Computer Vision Foundation / IEEE Computer Society, 2018, pp. 5265–5274.

[99] H. Masnadi-Shirazi and N. Vasconcelos, "Risk minimization, probability elicitation, and cost-sensitive svms," in *Proceedings of International Conference on Machine Learning (ICML)*, Madison, WI, USA: Omnipress, Jun. 2010, pp. 759–766.

[100] S. Khan, M. Hayat, S. W. Zamir, J. Shen, and L. Shao, "Striking the right balance with uncertainty," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2019, pp. 103–112.

[101] Y.-X. Wang, D. Ramanan, and M. Hebert, "Learning to model the tail," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, Long Beach, CA, USA: Curran Associates, Inc., Dec. 2017, pp. 7032–7042.

[102] C. Huang, Y. Li, C. C. Loy, and X. Tang, "Deep imbalanced learning for face recognition and attribute prediction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 11, pp. 2781–2794, 2019.

[103] Y. Wu, Q. Huang, Z. Liu, Y. Wang, and D. Lin, "Distribution-balanced loss for multi-label classification in long-tailed datasets," in *Proceedings of European conference on computer vision (ECCV)*, ser. Lecture Notes in Computer Science, vol. 12349, Glasgow, UK: Springer, Aug. 2020, pp. 162–178.

[104] X. Wang, L. Lian, Z. Miao, Z. Liu, and S. X. Yu, "Long-tailed recognition by routing diverse distribution-aware experts," in *Proceedings of International Conference on Learning Representations (ICLR)*, Virtual Event, Austria: OpenReview.net, May 2021. [Online]. Available: https://openreview.net/forum?id=D9I3drBz4UC.

[105] J. Cui, S. Liu, Z. Tian, Z. Zhong, and J. Jia, "Reslt: Residual learning for long-tailed recognition," *arXiv preprint arXiv:2101.10633*, 2021.

[106] M. A. Jamal, M. Brown, M.-H. Yang, L. Wang, and B. Gong, "Rethinking class-balanced methods for long-tailed visual recognition from a domain adaptation perspective," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2020, pp. 7610–7619.

[107] Y. Zhong, W. Deng, M. Wang, J. Hu, J. Peng, X. Tao, and Y. Huang, "Unequal-training for deep face recognition with long-tailed noisy data," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2019, pp. 7812–7821.

[108] R. Ranjan, C. D. Castillo, and R. Chellappa, "L2-constrained softmax loss for discriminative face verification," *arXiv preprint arXiv:1703.09507*, 2017. [Online]. Available: https://arxiv.org/pdf/1703.09507.pdf.

[109] S. M. Kakade, K. Sridharan, and A. Tewari, "On the complexity of linear prediction: Risk bounds, margin bounds, and regularization," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, Vancouver, British Columbia, Canada: Curran Associates, Inc., Dec. 2008, pp. 793–800.

[110] Y. Liu, H. Li, and X. Wang, "Rethinking feature discrimination and polymerization for large-scale recognition," *arXiv preprint arXiv:1710.00870*, 2017.

[111] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA: Computer Vision Foundation / IEEE Computer Society, Jul. 2017.

[112] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, Vancouver, BC, Canada: Curran Associates, Inc., Dec. 2019, pp. 8024–8035.

[113] E. Hoffer, I. Hubara, and D. Soudry, "Train longer, generalize better: Closing the generalization gap in large batch training of neural networks," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, Long Beach, CA, USA: Curran Associates, Inc., Dec. 2017, pp. 1731–1741.

[114] Y. Li, H. Zaragoza, R. Herbrich, J. Shawe-Taylor, and J. Kandola, "The perceptron algorithm with uneven margins," in *Proceedings of International Conference on Machine Learning (ICML)*, vol. 2, University of New South Wales, Sydney, Australia: Morgan Kaufmann, Jul. 2002, pp. 379–386.

[115] Y. Hong, S. Han, K. Choi, S. Seo, B. Kim, and B. Chang, "Disentangling label distribution for long-tailed visual recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Virtual Event: Computer Vision Foundation / IEEE Computer Society, Jun. 2021, pp. 6626–6636.

[116] A. Graves, G. Wayne, M. Reynolds, T. Harley, I. Danihelka, A. Grabska-Barwińska, S. G. Colmenarejo, E. Grefenstette, T. Ramalho, J. Agapiou, *et al.*, "Hybrid computing using a neural network with dynamic external memory," *Nature*, vol. 538, no. 7626, pp. 471–476, 2016.

[117] A. Graves, M. G. Bellemare, J. Menick, R. Munos, and K. Kavukcuoglu, "Automated curriculum learning for neural networks," in *Proceedings of International Conference on Machine Learning (ICML)*, ser. Proceedings of Machine Learning Research, vol. 70, Sydney, NSW, Australia: PMLR, Aug. 2017, pp. 1311–1320.

[118] C. Florensa, D. Held, M. Wulfmeier, M. Zhang, and P. Abbeel, "Reverse curriculum generation for reinforcement learning," in *Conference on robot learning*, PMLR, 2017, pp. 482–495.

[119] A. Pentina, V. Sharmanska, and C. H. Lampert, "Curriculum learning of multiple tasks," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA: Computer Vision Foundation / IEEE Computer Society, Jun. 2015.

[120] C. Liu, S. He, K. Liu, J. Zhao, *et al.*, "Curriculum learning for natural answer generation," in *International Joint Conferences on Artificial Intelligence (IJCAI)*, Stockholm, Sweden: ijcai.org, Jul. 2018, pp. 4223–4229.

[121] A. Jesson, N. Guizard, S. H. Ghalehjegh, D. Goblot, F. Soudan, and N. Chapados, "Cased: Curriculum adaptive sampling for extreme data imbalance," in *Medical Image Computing and Computer Assisted Intervention (MICCAI)*, ser. Lecture Notes in Computer Science, vol. 10435, Quebec City, QC, Canada: Springer International Publishing, Sep. 2017, pp. 639–646.

[122] Y. Yuan, K. Yang, J. Guo, C. Zhang, and J. Wang, "Feature incay for representation regularization," in *Proceedings of International Conference on Learning Representations (ICLR)*, Vancouver, BC, Canada: OpenReview.net, Apr. 2018.

[123] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne.," *Journal of machine learning research*, vol. 9, no. 11, 2008.

[124] B. Chen, W. Deng, and J. Du, "Noisy softmax: Improving the generalization ability of dcnn via postponing the early softmax saturation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA: Computer Vision Foundation / IEEE Computer Society, Jul. 2017, pp. 4021–4030.

[125] W. Zhang, Y. Chen, W. Yang, G. Wang, J.-H. Xue, and Q. Liao, "Class-variant margin normalized softmax loss for deep face recognition," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 10, pp. 4742–4747, 2021.

[126] L. B. Smith and L. K. Slone, "A developmental approach to machine learning?" *Frontiers in psychology*, vol. 8, p. 2124, 2017.

[127] L. Shen, Z. Lin, and Q. Huang, "Relay backpropagation for effective learning of deep convolutional neural networks," in *Proceedings of European conference on computer vision (ECCV)*, ser. Lecture Notes in Computer Science, vol. 9911, Amsterdam, The Netherlands: Springer, Oct. 2016, pp. 467–482.

[128] S. Ando and C. Huang, "Deep over-sampling framework for classifying imbalanced data," in *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD*, ser. Lecture Notes in Computer Science, vol. 10534, Skopje, Macedonia: Springer, Sep. 2017, pp. 770–785.

[129] P. Samira, T. Yudong, M. Anup, *et al.*, "Dynamic sampling in convolutional neural networks for imbalanced data classification," in *Proceeding of IEEE Conference on Multimedia Information Processing and Retrieval, MIPR*, Miami, FL, USA: IEEE Computer Society, Apr. 2018, pp. 112–117.

[130] C. Drummond, R. C. Holte, *et al.*, "C4.5, class imbalance, and cost sensitivity: Why under-sampling beats over-sampling," in *Workshop on learning from imbalanced datasets II*, vol. 11, 2003, pp. 1–8.

[131] H. Lee, M. Park, and J. Kim, "Plankton classification on imbalanced large scale database via convolutional neural networks with transfer learning," in *Proceedings of the IEEE International Conference on Image Processing, ICIP*, Phoenix, AZ, USA: IEEE Computer Society, Sep. 2016, pp. 3713–3717.

[132] S. H. Khan, M. Hayat, M. Bennamoun, F. A. Sohel, and R. Togneri, "Cost-sensitive learning of deep feature representations from imbalanced data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3573–3587, 2018.

[133] P. W. Koh and P. Liang, "Understanding black-box predictions via influence functions," in *Proceedings of International Conference on Machine Learning (ICML)*, vol. 70, Sydney, NSW, Australia: PMLR, Aug. 2017, pp. 1885–1894.

[134] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, May 2015.

[135] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA: Computer Vision Foundation / IEEE Computer Society, Jul. 2017, pp. 4700–4708.

[136] S. Li, K. Gong, C. H. Liu, Y. Wang, F. Qiao, and X. Cheng, "Metasaug: Meta semantic augmentation for long-tailed visual recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Virtual Event: Computer Vision Foundation / IEEE Computer Society, Jun. 2021, pp. 5212–5221.

[137] X. Wang, T. E. Huang, J. Gonzalez, D. Trevor, and F. Yu, "Frustratingly simple few-shot object detection," in *Proceedings of International Conference on Machine Learning (ICML)*, vol. 119, 2020, pp. 9919–9928.

[138] M. Andrychowicz, M. Denil, S. Gomez, M. W. Hoffman, D. Pfau, T. Schaul, B. Shillingford, and N. De Freitas, "Learning to learn by gradient descent by gradient

descent," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, Barcelona, Spain: Curran Associates, Inc., Dec. 2016, pp. 3981–3989.

[139] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proceedings of International Conference on Machine Learning (ICML)*, ser. Proceedings of Machine Learning Research, vol. 70, Sydney, NSW, Australia: PMLR, Aug. 2017, pp. 1126–1135.

[140] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[141] G. Chen, W. Choi, X. Yu, T. X. Han, and M. Chandraker, "Learning efficient object detection models with knowledge distillation," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, Curran Associates, Inc., 2017, pp. 742–751.

[142] T. Li, L. Wang, and G. Wu, "Self supervision to distillation for long-tailed visual recognition," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada: IEEE, Dec. 2021, pp. 630–639.

[143] T. Pang, K. Xu, and J. Zhu, "Mixup inference: Better exploiting mixup to defend adversarial attacks," in *Proceedings of International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia, Apr. 2020.

[144] L. Zhang, Z. Deng, K. Kawaguchi, A. Ghorbani, and J. Zou, "How does mixup help with robustness and generalization?" In *Proceedings of International Conference on Learning Representations (ICLR)*, Virtual Event, Austria, May 2021.

[145] Y.-m. Cheung and H. Zeng, "Local kernel regression score for selecting features of high-dimensional data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 12, pp. 1798–1802, 2009.

[146] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu, "Feature selection: A data perspective," *ACM Computing Surveys*, vol. 50, no. 6, pp. 1–45, Dec. 2017.

[147] Q. Peng, Y.-m. Cheung, X. You, and Y. Y. Tang, "A hybrid of local and global saliencies for detecting image salient region and appearance," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 1, pp. 86–97, 2017.

[148] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 10, pp. 1915–1926, 2012.

[149] N. Liu, N. Zhang, K. Wan, L. Shao, and J. Han, "Visual saliency transformer," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada: IEEE Computer Society, Oct. 2021, pp. 4702–4712.

[150] K. Han, A. Xiao, E. Wu, J. Guo, C. XU, and Y. Wang, "Transformer in transformer," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, vol. 34, Virtual Event: Curran Associates, Inc., Dec. 2021, pp. 15 908–15 919.

[151] T. Fawcett, "Roc graphs: Notes and practical considerations for researchers," *Machine learning*, vol. 31, no. 1, pp. 1–38, 2004.

[152] J. Li, R. Socher, and S. C. H. Hoi, "Dividemix: Learning with noisy labels as semi-supervised learning," in *Proceedings of International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia: OpenReview.net, Apr. 2020. [Online]. Available: https://openreview.net/forum?id=HJgExaVtwr.

[153] D. T. Nguyen, C. K. Mummadi, T. Ngo, T. H. P. Nguyen, L. Beggel, and T. Brox, "SELF: learning to filter noisy labels with self-ensembling," in *Proceedings of International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia: OpenReview.net, Apr. 2020. [Online]. Available: https://openreview.net/forum?id=HkgsPhNYPS.

[154] G. Pleiss, T. Zhang, E. R. Elenberg, and K. Q. Weinberger, "Identifying mislabeled data using the area under the margin ranking," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, Virtual Event: Curran Associates, Inc., Nov. 2020.

[155] P. Vincent, H. Larochelle, Y. Bengio, and P. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of International Conference on Machine Learning (ICML)*, ser. ACM International Conference Proceeding Series, vol. 307, Helsinki, Finland: ACM, Jun. 2008, pp. 1096–1103. [Online]. Available: https://doi.org/10.1145/1390156.1390294.

[156] J. Batson and L. Royer, "Noise2self: Blind denoising by self-supervision," in *Proceedings of International Conference on Machine Learning (ICML)*, ser. Proceedings of Machine Learning Research, vol. 97, Long Beach, California, USA: PMLR, Jun. 2019, pp. 524–533. [Online]. Available: http://proceedings.mlr.press/v97/batson19a.html.

[157] S. Gidaris, A. Bursuc, N. Komodakis, P. P. érez, and M. Cord, "Boosting few-shot visual learning with self-supervision," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Seoul, Korea (South): IEEE Computer Society, Oct. 2019, pp. 8058–8067. [Online]. Available: https://doi.org/10.1109/ICCV.2019.00815.

[158] C. Chuang, J. Robinson, Y. Lin, A. Torralba, and S. Jegelka, "Debiased contrastive learning," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, Virtual Event: Curran Associates, Inc., Nov. 2020.

[159] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," in *Proceedings of Neural Information Processing Systems (NeurIPS)*, Virtual Event: Curran Associates, Inc., Dec. 2020.

CURRICULUM VITAE

Academic qualification of the thesis author, Ms. LI Mengke:

- Received the degree of Bachelor of Engineering from Hanhong College, Southwest University, June 2015.
- Received the degree of Master of Engineering from School of Electronic Engineering, Xidian University, June 2018.

May 2022

LMK thesis

ORIGINALITY REPORT

23%	18%	12%	12%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	web.archive.org Internet Source	3%
2	arxiv.org Internet Source	1%
3	www.sciencegate.app Internet Source	1%
4	Submitted to Aberystwyth University Student Paper	1%
5	repository.hkbu.edu.hk Internet Source	1%
6	deepai.org Internet Source	1%
7	arxiv-export-lb.library.cornell.edu Internet Source	1%
8	Zhisheng Zhong, Jiequan Cui, Shu Liu, Jiaya Jia. "Improving Calibration for Long-Tailed Recognition", 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021 Publication	1%

9	Submitted to Middle East Technical University Student Paper	1 %
10	Yan Wang, Yongshun Zhang, Furao Shen, Jian Zhao. "Dynamic Auxiliary Soft Labels for decoupled learning", Neural Networks, 2022 Publication	<1 %
11	www.hindawi.com Internet Source	<1 %
12	"Computer Vision – ECCV 2020", Springer Science and Business Media LLC, 2020 Publication	<1 %
13	export.arxiv.org Internet Source	<1 %
14	Submitted to Hong Kong Baptist University Student Paper	<1 %
15	Boyan Zhou, Quan Cui, Xiu-Shen Wei, Zhao-Min Chen. "BBN: Bilateral-Branch Network With Cumulative Learning for Long-Tailed Visual Recognition", 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020 Publication	<1 %
16	www.mdpi.com Internet Source	<1 %
17	Submitted to University of Sheffield Student Paper	<1 %

18	Submitted to Middle Tennessee State University Student Paper	<1 %
19	Submitted to University of Oxford Student Paper	<1 %
20	Submitted to Mae Fah Luang University Student Paper	<1 %
21	Submitted to University of Glasgow Student Paper	<1 %
22	Submitted to University of Wollongong Student Paper	<1 %
23	www.cs.toronto.edu Internet Source	<1 %
24	Submitted to University of Queensland Student Paper	<1 %
25	Xiaokang Zhang, Yuanlue Zhu, Wenting Chen, Wenshuang Liu, Linlin Shen. "Gated SwitchGAN for Multi-Domain Facial Image Translation", IEEE Transactions on Multimedia, 2022 Publication	<1 %
26	Submitted to University College London Student Paper	<1 %
27	Jialun Liu, Yifan Sun, Yijin Xu, Hongbin Pei, Wenhui Li. "Feature Cloud: Improving Deep	<1 %

Visual Recognition with Probabilistic Feature Augmentation", IEEE Transactions on Circuits and Systems for Video Technology, 2021
Publication

28	Submitted to Macau University of Science and Technology Student Paper	<1%
29	Submitted to The Hong Kong Polytechnic University Student Paper	<1%
30	Jiaqi Wang, Wenwei Zhang, Yuhang Zang, Yuhang Cao, Jiangmiao Pang, Tao Gong, Kai Chen, Ziwei Liu, Chen Change Loy, Dahua Lin. "Seesaw Loss for Long-Tailed Instance Segmentation", 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021 Publication	<1%
31	Submitted to Institute of Technology Blanchardstown Student Paper	<1%
32	Submitted to Korea University Student Paper	<1%
33	Submitted to University of Colorado, Denver Student Paper	<1%
34	Qiong Chen, Qingfa Liu, Enlu Lin. "A knowledge-guide hierarchical learning method	<1%

for long-tailed image classification", Neurocomputing, 2021
Publication

35	Submitted to University of Warwick Student Paper	<1 %
36	Shaoyu Zhang, Chen Chen, Xiujuan Zhang, Silong Peng. "Label-Occurrence-Balanced Mixup for Long-Tailed Recognition", ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2022 Publication	<1 %
37	zenodo.org Internet Source	<1 %
38	Submitted to University of Durham Student Paper	<1 %
39	boqinggong.info Internet Source	<1 %
40	github.com Internet Source	<1 %
41	Submitted to Özyegin Üniversitesi Student Paper	<1 %
42	ebin.pub Internet Source	<1 %
43	Submitted to University of Wales Swansea Student Paper	<1 %

44	www.mikeprocopio.com Internet Source	<1 %
45	yibolin.com Internet Source	<1 %
46	Hongyu Wang, Henry Gouk, Huon Fraser, Eibe Frank, Bernhard Pfahringer, Michael Mayo, Geoffrey Holmes. "Experiments in cross-domain few-shot learning for image classification", Journal of the Royal Society of New Zealand, 2022 Publication	<1 %
47	Submitted to University of Sydney Student Paper	<1 %
48	Submitted to Indian Institute of Technology Guwahati Student Paper	<1 %
49	Submitted to National Taipei University of Education Student Paper	<1 %
50	Submitted to Imperial College of Science, Technology and Medicine Student Paper	<1 %
51	Yanran Wang, Qingliang Chen, Shilang Chen, Junjun Wu. "Multi-Scale Convolutional Features Network for Semantic Segmentation in Indoor Scenes", IEEE Access, 2020 Publication	<1 %

52	"Computer Vision – ECCV 2020 Workshops", Springer Science and Business Media LLC, 2020 Publication	<1 %
53	Submitted to University of Nevada Reno Student Paper	<1 %
54	mdpi-res.com Internet Source	<1 %
55	www.fst.umac.mo Internet Source	<1 %
56	Chi Zhang, Benyi Hu, Yuhang Liuzhang, Le Wang, Li Liu, Yuehu Liu. "Switching: understanding the class-reversed sampling in tail sample memorization", Machine Learning, 2022 Publication	<1 %
57	Peng Wang, Kai Han, Xiu-Shen Wei, Lei Zhang, Lei Wang. "Contrastive Learning based Hybrid Networks for Long-Tailed Image Classification", 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021 Publication	<1 %
58	Submitted to The University of Manchester Student Paper	<1 %
59	Submitted to University of Birmingham Student Paper	

		<1%
60	Submitted to University of Ulster Student Paper	<1%
61	ro.ecu.edu.au Internet Source	<1%
62	ruor.uottawa.ca Internet Source	<1%
63	Submitted to University of Exeter Student Paper	<1%
64	Submitted to University of Lancaster Student Paper	<1%
65	www.comp.hkbu.edu.hk Internet Source	<1%
66	Submitted to University of Edinburgh Student Paper	<1%
67	Zonghai Zhu, Huanlai Xing, Yuge Xu. "Easy balanced mixing for long-tailed data", Knowledge-Based Systems, 2022 Publication	<1%
68	Muhammad Abdullah Jamal, Matthew Brown, Ming-Hsuan Yang, Liqiang Wang, Boqing Gong. "Rethinking Class-Balanced Methods for Long-Tailed Visual Recognition From a Domain Adaptation Perspective", 2020	<1%

IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020
Publication

69	Xiangxian Li, Haokai Ma, Lei Meng, Xiangxu Meng. "Comparative Study of Adversarial Training Methods for Long-tailed Classification", Proceedings of the 1st International Workshop on Adversarial Learning for Multimedia, 2021 Publication	<1 %
70	centaur.reading.ac.uk Internet Source	<1 %
71	150.214.191.180 Internet Source	<1 %
72	Submitted to Vrije Universiteit Brussel Student Paper	<1 %
73	Cheng Zhang, Wan Shou Jiang, Yuan Zhang, Wei Wang, Qing Zhao, Chen Jie Wang. "Transformer and CNN Hybrid Deep Neural Network for Semantic Segmentation of Very-high-resolution Remote Sensing Imagery", IEEE Transactions on Geoscience and Remote Sensing, 2022 Publication	<1 %
74	eprints.ecs.soton.ac.uk Internet Source	<1 %

papers.neurips.cc

75	Internet Source	<1%
76	Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, Serge Belongie. "Class-Balanced Loss Based on Effective Number of Samples", 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019 Publication	<1%
77	rdw.rowan.edu Internet Source	<1%
78	Jaehyung Kim, Jongheon Jeong, Jinwoo Shin. "M2m: Imbalanced Classification via Major-to-Minor Translation", 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020 Publication	<1%
79	Junjie Zhang, Lingqiao Liu, Peng Wang, Jian Zhang. "Exploring the auxiliary learning for long-tailed visual recognition", Neurocomputing, 2021 Publication	<1%
80	d-nb.info Internet Source	<1%
81	goldberg.berkeley.edu Internet Source	<1%
82	pure.uva.nl Internet Source	<1%

83	Submitted to Boston University Student Paper	<1%
84	Submitted to Bridgepoint Education Student Paper	<1%
85	Shuang Li, Kaixiong Gong, Chi Harold Liu, Yulin Wang, Feng Qiao, Xinjing Cheng. "MetaSAug: Meta Semantic Augmentation for Long-Tailed Visual Recognition", 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021 Publication	<1%
86	Submitted to Universidad de Alcalá Student Paper	<1%
87	downloads.hindawi.com Internet Source	<1%
88	Submitted to King's College Student Paper	<1%
89	Submitted to University of Technology, Sydney Student Paper	<1%
90	www.arxiv-vanity.com Internet Source	<1%
91	Submitted to Monash University Student Paper	<1%
92	Submitted to University of Cape Town	

	Student Paper	<1%
93	Submitted to University of Illinois at Urbana-Champaign Student Paper	<1%
94	www.nii.ac.jp Internet Source	<1%
95	Keqi Deng, Gaofeng Cheng, Runyan Yang, Yonghong Yan. "Alleviating ASR Long-tailed Problem by Decoupling the Learning of Representation and Classification", IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2021 Publication	<1%
96	Submitted to Cornell University Student Paper	<1%
97	Ruru Zhang, Haihong E, Lifei Yuan, Jiawen He, Hongxing Zhang, Shengjuan Zhang, Yanhui Wang, Meina Song, Lifei Wang. "MBNM: Multi-branch network based on memory features for long-tailed medical image recognition", Computer Methods and Programs in Biomedicine, 2021 Publication	<1%
98	Songyang Zhang, Zeming Li, Shipeng Yan, Xuming He, Jian Sun. "Distribution Alignment: A Unified Framework for Long-tail Visual	<1%

Recognition", 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021
Publication

99	scholarcommons.usf.edu Internet Source	<1%
100	Submitted to Babes-Bolyai University Student Paper	<1%
101	Hsin-Ping Chou, Shih-Chieh Chang, Jia-Yu Pan, Wei Wei, Da-Cheng Juan. "Chapter 9 Remix: Rebalanced Mixup", Springer Science and Business Media LLC, 2020 Publication	<1%
102	Submitted to International Institute of Information Technology, Hyderabad Student Paper	<1%
103	www.acberg.com Internet Source	<1%
104	Submitted to City University of Hong Kong Student Paper	<1%
105	Tong Wu, Ziwei Liu, Qingqiu Huang, Yu Wang, Dahua Lin. "Adversarial Robustness under Long-Tailed Distribution", 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021 Publication	<1%

106	spectrum.library.concordia.ca Internet Source	<1 %
107	stars.library.ucf.edu Internet Source	<1 %
108	123dok.co Internet Source	<1 %
109	Submitted to Queen's University of Belfast Student Paper	<1 %
110	Linchao Zhu, Yi Yang. "Inflated Episodic Memory With Region Self-Attention for Long-Tailed Visual Recognition", 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020 Publication	<1 %
111	Submitted to National Research University Higher School of Economics Student Paper	<1 %
112	Submitted to University of Bristol Student Paper	<1 %
113	Submitted to University of Leeds Student Paper	<1 %
114	Submitted to University of Westminster Student Paper	<1 %
115	orca.cf.ac.uk Internet Source	<1 %

116	www.schuller.it Internet Source	<1 %
117	Jianfeng Wang, Thomas Lukasiewicz, Xiaolin Hu, Jianfei Cai, Zhenghua Xu. "RSG: A Simple but Effective Module for Learning Imbalanced Datasets", 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021 Publication	<1 %
118	repository.kulib.kyoto-u.ac.jp Internet Source	<1 %
119	www.new-npac.org Internet Source	<1 %
120	www.patentgenius.com Internet Source	<1 %
121	Submitted to University of Newcastle Student Paper	<1 %
122	archiv.ub.uni-marburg.de Internet Source	<1 %
123	seat.massey.ac.nz Internet Source	<1 %
124	trepo.tuni.fi Internet Source	<1 %
125	Submitted to Heriot-Watt University Student Paper	<1 %

126	Submitted to Innopolis University Student Paper	<1 %
127	Yuhang Zang, Chen Huang, Chen Change Loy. "FASA: Feature Augmentation and Sampling Adaptation for Long-Tailed Instance Segmentation", 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 2021 Publication	<1 %
128	cmpe.emu.edu.tr Internet Source	<1 %
129	digital.library.adelaide.edu.au Internet Source	<1 %
130	digital.library.unt.edu Internet Source	<1 %
131	doras.dcu.ie Internet Source	<1 %
132	Jialian Wu, Liangchen Song, Tiancai Wang, Qian Zhang, Junsong Yuan. "Forest R-CNN: Large-Vocabulary Long-Tailed Object Detection and Instance Segmentation", Proceedings of the 28th ACM International Conference on Multimedia, 2020 Publication	<1 %
133	Jiarui Cai, Yizhou Wang, Jenq-Neng Hwang. "ACE: Ally Complementary Experts for Solving Long-Tailed Recognition in One-Shot", 2021	<1 %

	IEEE/CVF International Conference on Computer Vision (ICCV), 2021 Publication	
134	Submitted to Rochester Institute of Technology Student Paper	<1 %
135	Tianhao Li, Limin Wang, Gangshan Wu. "Self Supervision to Distillation for Long-Tailed Visual Recognition", 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 2021 Publication	<1 %
136	Xiyan Deng, Yuping Wang, Minqi Li. "A New Classifier by Remixing Features and Adjusting Logits for Long-Tailed Visual Data", 2021 17th International Conference on Computational Intelligence and Security (CIS), 2021 Publication	<1 %
137	eprints-phd.biblio.unitn.it Internet Source	<1 %
138	ethesis.nitrkl.ac.in Internet Source	<1 %
139	hub.hku.hk Internet Source	<1 %
140	res.mdpi.com Internet Source	<1 %

141	www.cs.put.poznan.pl Internet Source	<1%
142	www.google.it Internet Source	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		