Neurocomputing 509 (2022) 290–309

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Survey paper

Long-tailed visual recognition with deep models: A methodological survey and evaluation ☆

Yu Fu [a], Liuyu Xiang [b], Yumna Zahid [a], Guiguang Ding [b], Tao Mei [c], Qiang Shen [a], Jungong Han [a,*]

[a] Aberystwyth University, Computer Science Department, Aberystwyth SY23 3DB, Ceredigion, UK
[b] Tsinghua University, School of Software, Beijing, China
[c] JD, AI Research, Beijing, China

ARTICLE INFO

Article history:
Received 14 January 2022
Revised 19 May 2022
Accepted 3 August 2022
Available online 13 August 2022

Keywords:
Long-tail
Visual recognition
Deep learning

ABSTRACT

In the real world, large-scale datasets for visual recognition typically exhibit a long-tailed distribution, where only a few classes contain adequate samples but the others have (much) fewer samples. With the advancement of data-hungry deep models for visual recognition, the low-tail power-law data distribution that biases the model training has attracted significant attention. When training with the long-tailed data, the majority classes dominate the training procedure, resulting in poor performance in instance-scarce classes. To tackle this problem, numerous strategies, such as re-sampling, cost-sensitive loss, meta-learning and transfer learning, have been proposed. This paper systematically reviews contemporary approaches for the long-tailed visual recognition task and categorizes these methods based on the stage applied as training, fine-tuning, and inference. Furthermore, we categorize training stage methods into data augmentation, re-sampling strategy, cost-sensitive loss, as well as multiple experts and transfer learning. Next, comprehensive comparisons are made in the balanced test set performance of long-tailed benchmarks and method robustness in diverse test distributions using metrics including top-1 accuracy, per-class accuracy, multi-class ROC AUC and Expected Calibration Error (ECE). At last, we outline the challenges in this field and future research trends. Our reviews and intriguing findings can be a tutorial for researchers working in the field of open-world deep learning.

1. Introduction

A long-tailed distribution, also known as a lower-tail power-law distribution [100], is where the frequency of many events or classes is much lower compared to the few others. Such phenomenon is prevalent in the real-world scenarios such as disparity in incomes, sand particle sizes, meteor impacts on the moon, frequencies of words in a text [76]. Likewise, in the realm of deep learning for visual recognition, the real-world image datasets also exhibit long-tailed distribution as evidenced by widely used iNaturalist [35], and LVIS [28] datasets. As shown in Fig. 1, when the categories in the iNaturalist dataset are organized in a descending order of frequency, high-frequency classes are followed by a population of low-frequency classes which gradually 'tail off'.

Visual recognition has made rapid advances with the development of deep learning. However, it is well known that deep learning is data-hungry, and both the quantity and quality of the training data determine the model performance. When deep learning meets long-tailed datasets during training, it will learn a biased model since the head classes dominate the parameter optimization, resulting in low performance for the tail classes. Although an intuitive solution is to balance training set in real scenarios, it is highly time-consuming and requires commercial expense, especially for data-poor species.

Not surprisingly, the importance of the long-tailed visual recognition problem in deep neural network training is becoming increasingly recognized by researchers. In recent years, extensive research has been presented to deal with long-tailed visual recognition problems, including re-sampling methods [4,40,62,69], cost-sensitive methods [12,34,37,38,51,105] and multiple experts [106,116,118,132]. The increasing number of publications devoted to the long-tailed visual problem, as illustrated in Fig. 2, demonstrates the problem's growing prominence in the last 5 years.

Although there are several survey papers proposed in the field of imbalanced learning [2,4,25,29,30,45,71,90], the systematically reviewed literature in long-tailed visual recognition with deep models is limited. Zhang et al. [128] recently revisited simple but

☆ This work is partly supported by the British Council under Grant No.: 623718881.

* Corresponding author.
E-mail address: juh22@aber.ac.uk (J. Han).

https://doi.org/10.1016/j.neucom.2022.08.031
0925-2312/

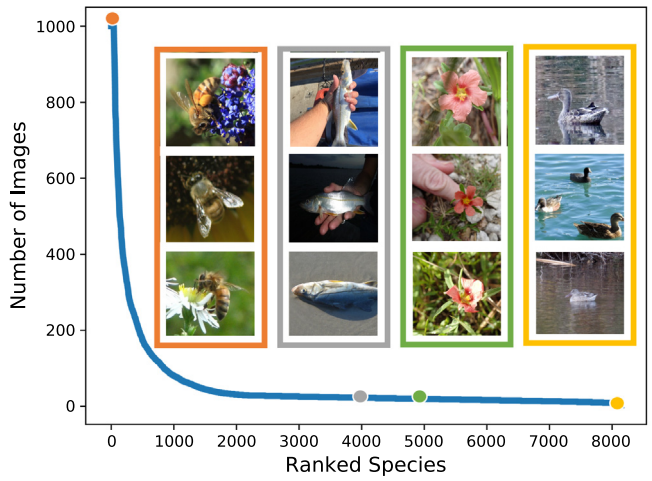

Fig. 1. Long-tailed distribution of iNaturalist 2018 dataset. X-axis and y-axis show the ranked species by frequency and the number of images. Class instances are shown in frames with different colors, which correspond to the points in the line.

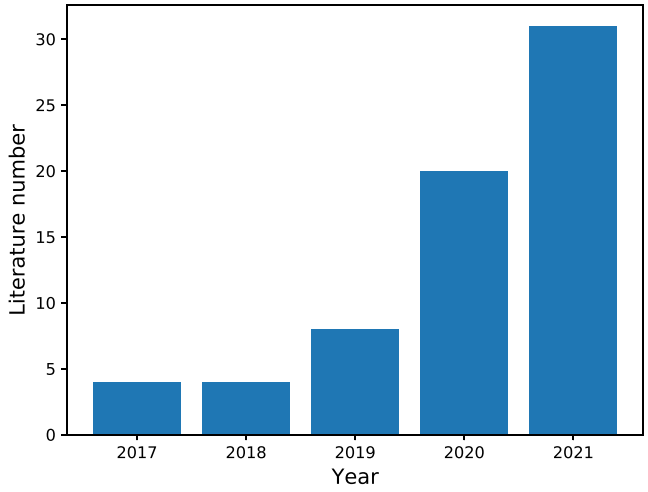

Fig. 2. Number of publications devoted to solving the long-tailed visual problem over the past 5 years.

effective strategies such as re-sampling and data augmentation methods. However, this study does not engage with the discussion on the sophisticated strategies and the taxonomy in long-tailed recognition methods.

To the best of our knowledge, this is the first study that aims to identify and evaluate methods systematically for long-tailed visual recognition. We provide a thorough discussion on deep long-tailed visual recognition methods, presenting both simple yet effective and complex strategies. These strategies are grouped according to their applicability in the deep-learning stages, i.e., training, fine-tuning, and inference. The training stage techniques are further classified into sub-categories: data augmentation, re-sampling, cost-sensitive loss, and multiple experts and transfer learning. We also provide an extensive method comparison using different evaluation approaches in test set with diverse distributions.

Our key contributions are as follows: 1) We provide a comprehensive discussion on long-tailed visual recognition techniques with deep-learning models. 2) The taxonomy of methods is arranged according to at which stage of deep learning the contributed modules can help. 3) We compare the results using various test set distributions and metrics on multiple benchmark datasets.

The rest of this study is structured as follows (Fig. 3): Section 2 explores background information on closely related work such as imbalanced learning, deep representation and few-shot learning. Discussion on datasets, evaluation metrics are provided in Section 3. The state-of-the-art works categorized into training, fine-tuning and inference stages are discussed in Section 4. Qualitative and quantitative studies are conducted and evaluated based on benchmark datasets in Section 5. Finally, we summarize the challenges, outline the major research trends and conclude the paper in Section 6.

2. Background

In this section, we investigate four interleaved problems: imbalanced learning, deep representation learning, fine-grained recognition, and few-shot learning.

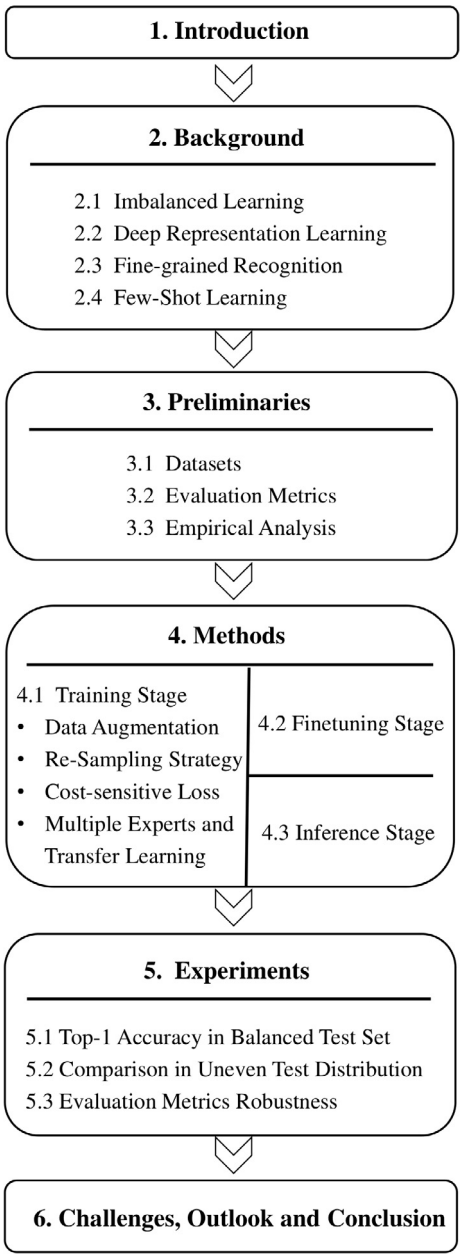

Fig. 3. Long-tailed survey summary.

2.1. Imbalanced learning

The imbalanced distribution means that one or a few classes own many more instances by the ratio of 100:1, 1000:1 even 10,000:1 to others [25,29]. The imbalanced learning in data mining and data classification is mostly related to rare events (REs), including software defects [81], cancer gene expression [122] and natural disasters [61]. In those scenarios, the REs are highly significant, and misclassification would result in high costs. For instance, the incorrect classification in predicting natural disasters might cause massive loss of people and property. Therefore, a series of strategies are proposed to alleviate imbalance influence, including re-sampling methods [7,65], cost-sensitive strategies [63,89,97], kernel-based methods [32,113,114] etc. In recent years, deep learning methods have achieved remarkable progress in various computer vision tasks. However, deep model training usually requires a large number of samples associated with numerous categories, making the imbalanced problem severe and more challenging. The imbalanced learning in this context has gained increasing attention in recent years and researchers usually refer to such a case as long-tailed learning.

2.2. Deep representation learning

Data representation generally determines the performance of models, and therefore, much effort goes into designing processing pipelines and input transformations to learn discriminative information for efficient machine learning [1]. The development of deep neural networks (DNNs) with layered incremental feature learning has provided useful and easier data representations. One of the most popular DNNs, called ResNet [31], adds shortcut connections to several stacked layers to deal with the accuracy degradation problem when networks go deeper. ResNet and its variant ResNeXt [120] have achieved distinguished performance in detection, localization and segmentation tasks and have become the backbone networks for long-tailed vision recognition tasks. There are several versions of ResNet/ResNeXt with 10, 32, 50, 101 layers in five blocks. When training with high-complexity deep models, large scale training datasets are acquired to avoid over-fitting. These models are trained on popular visual recognition benchmarks such as CIFAR [46] and ImageNet [15] that have balanced distributions. However, a considerable degradation of performance, especially in data-scarcity classes, is observed from DNNs trained in long-tailed datasets. Therefore, investigating the long-tailed visual problems is significant so as to adapt deep models to real-world scenarios.

2.3. Fine-grained recognition

With the development of deep learning techniques, fine-grained image recognition, which deals with recognizing objects to sub-categories of the same meta-categories (e.g., bird, dogs, cars), has been an active area in recent years. In reality, fine-grained image recognition has a wide range of applications, such as biodiversity monitoring and climate change evaluation [112]. This task is challenging because the methods need to localize and represent the marginal visual differences within sub-categories [129]. Therefore, fine-grained approaches usually carry out two steps: fine-grained features learning [14,18,44,52] and discriminative part localization [21,111,125,129]. In real-world tasks, the long-tailed and fine-grained visual recognition are always interleaved and appear simultaneously. A simple example is that the long-tailed recognition in existing datasets, e.g., ImageNet-LT, iNaturalist, and LVIS, already encountered fine-grained recognition problems where many tail and head classes are inherited from the same root meta-category. Likewise, some fine-grained datasets, such as Morph II and FG-NET, are long-tailed. The fine-grained classes make the long-tailed problem harder, where the tail classes have a higher probability of being classified to similar head classes. The combination of fine-grained and long-tailed methods should be explored to discriminate differences between fine-grained categories in the imbalanced setting.

2.4. Few-shot learning

To deal with the lack of labeled training examples, few-shot learning (FSL) is emerged. It quickly generalises the models to new tasks using a small number of annotated samples and prior knowledge [108]. How to use the prior knowledge and those few samples would determine the method performance. One of the earliest examples is of mimicking human cognitive ability to parse and generate new handwritten characters using a few examples [47]. A solution is to decompose the characters into transferable parts and then assemble these components into new characters. Meta learning, or learning to learn, is one commonly used technique for few-shot learning tasks. It is designed for the model to generalize across tasks so that the model can quickly adapt to few-shot learning tasks. [23,75,86,87,91]. From this perspective, few-shot learning could be applied to the long-tailed recognition problems where the tail classes have few samples.

3. Preliminaries

Formally, we define the long-tailed visual recognition dataset as image data having a long-tailed distribution for recognition, where head classes are associated with significantly more samples than tailed classes.

The general visual recognition datasets as well as other fine-grained recognition datasets are introduced in this section. We also define the causes of the long-tailed problem, followed by an empirical evaluation of classifier weight and accuracy bias. Evaluation metrics to measure the model performance for different objectives and notations are presented as well.

3.1. Datasets

The long-tailed datasets are made up of samples that were obtained naturally or manually selected based on the exponential distribution. In this section, the general object datasets, CIFAR-LT [12], ImageNet-LT [60], iNaturalist [35] and fine-grained datasets including Morph II [80], FG-NET [72], ChaLearn LAP 2015 [20], IMDB-WIKI [82], Places-LT [60], SUN-LT, MS1M-LT [60,109], CUB-LT [84] and AWA-LT [84] will be introduced. Unlike the general object datasets, the fine-grained datasets are subordinate categories with small inter-class variations and larger intra-class variations. Table 1 summarizes the dataset.

3.1.1. General object datasets

- **CIFAR-LT**: The original CIFAR dataset has two versions: CIFAR-10 and CIFAR-100. The former has ten classes, 6000 images in every class, while the latter contains 100 classes with 600 examples in each category [46]. Based on the evenly distributed original CIFAR dataset, the long-tailed versions are created and become benchmark datasets [12]. These datasets contain the same categories as the original one, however, the number of samples in each class is reduced based on equation $n = n_t \times \mu^t$ to around 12,000 images, where t is the class index counted from index 0 and n_t is the original class number with $\mu \in (0, 1)$. The test set remains unchanged with even distribution. The imbalanced factor (IF) of the long-tailed dataset is calculated by the equation

Table 1
The details of Long-tailed image recognition datasets.

Datasets	Fields	Categories	Training Samples
CIFAR-LT-10	General	10	12,406(im100)/13,996(im50)
CIFAR-LT-100	General	100	10,847(im100)/12,608(im50)
ImageNet-LT	General	1000	115,846
iNaturalist 2018	General	8142	437,513
Morph II	Age	62	10,634(S1-S2-S3)/4,395(80–20)
FG-NET	Age	70	1,002
ChaLearn LAP 2015	Age	101	2,476
IMDB-WIKI	Age	101	297,163
Places-LT	Scene	365	62,500
SUN	Scene	397	35,018(SUN-397)/4,084(SUN-LT)
MS1M-LT	Face Recognition	74,500	887,530
CUB-LT	Bird	200	2,945
AWA-LT	Animal	50	6,713

$IF = n_l/n_s$ where n_l is the largest class number and n_s is the smallest class number. The most common *IFs* are 50 and 100.

- **ImageNet-LT**: The ImageNet dataset is an image dataset constructed based on the WordNet structure. Visual competitions based on this dataset make a significant contribution to the development of computer vision [15]. ImageNet has 1000 classes with uniform distribution and each contains 1300 images. From this base dataset, the long-tailed version is constructed via Pareto distribution with the power value $\alpha = 6$ maximum 1280 samples and the minimum 5 samples [60]. Similar to CIFAR-LT, the test set is balanced. The ImageNet-LT dataset is gathered into three groups: many shot, median shot, and low shot with the cardinality thresholds 100 and 20. The class numbers in those are 391, 463, and 146, respectively.
- **iNaturalist**: The iNaturalist dataset is created because most of the image classification datasets have unnatural distribution. It proves that the natural world is heavily unbalanced where some classes are common and easy to photograph than others [98]. This dataset is based on the iNatrualist where naturalists share and map their observations of biodiversity around the world, contributing to the long-tailed feature. The 2018 version of iNaturalist dataset contains 8142 species and 437,513 images and could be directly used as a backbone dataset in long-tailed visual classification.

3.1.2. Fine-grained datasets

- **Morph II**: This dataset is a grayscale face dataset containing 55,134 face images collected from 13,617 individuals with the age range from 16 to 77. This dataset is used in two types of settings, as detailed in [94]: 1) S1-S2-S3 protocol, where three non-overlapping subsets are created. Experiments are repeated for all random combinations of two sets for testing and one for training. 2) 80–20 protocol, in which, to reduce the cross-race bias, a subset of 5,493 Caucasian descent images are used and further divided into 80% training and 20% testing sets.
- **FG-NET**: The FG-NET dataset has 1,002 face images from 82 subjects with 12 samples per subject. The age ranges from 0 to 69. The leave-one-out (LOPO) setting could be used to generate an evaluation set. Specifically, in LOPO, the test set is constructed from images belonging to a single person using the random sampling method [17].
- **ChaLearn LAP**: The ChaLearn LAP dataset contains facial images with labels according to the visible age, and is constructed for ChaLearn LAP challenge. The version of 2015 is commonly used as a long-tailed dataset [17]. Different from previous age datasets, it has no true age annotations but is set by the average age marked by ten people. The sample size of the training set, validation set and test set are 2476, 1136 and 1079, respectively.
- **IMDB-WIKI**: The IMDB-WIKI dataset is built up for the age estimation task with the images crawled from IMDb and Wikipedia. It contains 523,051 training face images where only 5% of objectives own more than 100 images whereas the average is 23 per class [82]. This dataset exhibits long-tailed distribution naturally.
- **Places-LT**: The Places-LT is artificially constructed from dataset Places 365 standard [60], which is a subset of the Places database with 434 place categories. The Place 365 standard dataset is a relatively balanced dataset with a minimum cardinality of 3068 and a maximum of 5000. The original validation set and test set are balanced with 50 images and 900 images accordingly. The long-tailed training data were generated with power value $\alpha = 6$ Pareto distribution with the category cardinality from 5 to 4980 in 365 classes.
- **SUN 397 (SUN-LT)**: The Scene UNderstanding (SUN) dataset is constructed with the selected 397 WordNet environmental, well-sampled categories containing from 100 to 2361 unique photographs each class. This dataset could be directly treated as a backbone dataset for the natural long-tailed distribution [109]. Also, a long-tail subset SUN-LT is constructed, which consists of 4,084 training images with attributes [84].
- **MS1M-LT**: A refined MS1M-ArcFace based on a large-scale dataset MS-Celeb-1M is proposed with 85 K identities and 5.8 M images [16,27]. Liu et al. [60] constructed a long-tailed version of the face recognition dataset containing 887.5 K images and 74.5 K identities. As for the test set, they introduced MegaFace, which has 3,530 images in the probe set and 1M images in the gallery set. The objective is to match each sample in prob set to gallery set with top-1 accuracy. Many-shot ($\geqslant 5$), few-shot ($< 5\ \&\ \geqslant 2$), one-shot ($< 2\ \&\ \geqslant 1$) and zero-shot ($= 0$) subsets are generated by pseudo occurrences through counting the similar (similarity greater than 0.7) training samples.
- **CUB-LT**: The balanced CUB-200–2011 dataset [101] is a bird recognition dataset containing 200 species and 11,788 images constructed in 2011. The purpose of this dataset is to facilitate the bird classification research where many bird species are visually indistinguishable. The species list is obtained from the online field guide, and the images are constructed in the Flickr image search and filtered by users of Mechanical Turk. Based on the CUB-200–2011 dataset, CUB-LT is built up according to the exponentially decaying function $f(class) = a, b^{-rank(class)}$ where the values of a, b meet the demand that the first class has the maximum samples while the lowest class has the sample size of 2 to 3. The validation set is balanced and accounts for 20% percentage of the training data [84].
- **AWA-LT**: The Animal with Attributes (AWA) dataset [48] contains 50 animal classes and 30,475 images with a minimum of 92 images per class. These images are collected from Google, Microsoft, Yahoo and Flickr search energies. The long-tailed ver-

sion of AWA dataset is established according to the same principle with CUB-LT. It consists of 6,713 training instances with 2 to 720 images for each class [84].

3.2. Evaluation metrics

Many evaluation metrics are proposed to assess the performance of the methodologies in different recognition tasks such as object recognition, age classification and face recognition. In this section, accuracy, precision, recall, F1 score, per-class accuracy, long-tailed accuracy, mean absolute error (MAE), ϵ-error, multi-class ROC AUC and expected calibration error (ECE) are introduced.

- **Accuracy, Precision, Recall and F1 score**: Accuracy, precision, recall and F1 are the basic metrics used to quantify performance on the retrieved data from sample space in tasks ranging from pattern recognition to image classification.
  Accuracy is the measure of correctly labeled instances from all test samples. This is also known as top-1 accuracy. Due to the long-tailed distribution, simple accuracy proves to be a naive measure of performance as it can be easily affected by multiple factors. Precision measures the number of correct positive labeled instances against all positive labeled samples while recall measures the number of correct positive labeled against actual positive labeled. F1 measure strikes a balance between the two by measuring their harmonic mean [3]. To provide a holistic comparison of long-tailed distribution methodologies, we take top-1 accuracy into account to evaluate the performance.
- **Per-Class Accuracy** Acc_{PC}: The per-class accuracy evaluation metric calculates the accuracy of each class separately and gets the final average sum [84]. The formula is $Acc_{PC} = \frac{1}{C}\sum_{c=1}^{C} Acc(c)$, where the $Acc(c)$ is the accuracy of class c. This method treats each class equally.
- **Long-Tailed Accuracy** Acc_{LT}: The long-tailed accuracy is used when the test distribution is long-tailed. Tackling the Acc_{LT} in an approximate uniform test set, it is proposed to take a weighted sum based on training set [84]. Specifically, the weight $p_{train}(c)$ is generated according to the LT distribution of training set such that $0 < p_{train}(c) < 1$ and $\sum_c p_{train} = 1$. Then the $ACC_{LT} = \sum_{c=1}^{C} p_{train}(c)Acc(c)$.
- **Many-Shot, Median-Shot, Low-Shot Accuracy**: Liu et al. [60] proposed two sample size thresholds, 80 and 20, to categorize the LT training set. Specifically, three subsets are many-shot ($frequency < 100$), median-shot ($20 \leqslant frequency \leqslant 100$) and low-shot ($frequency < 20$). The Acc_{MS}, Acc_{MED} and Acc_{LS} are the accuracy in three subsets.
- **Mean Absolute Error (MAE)** [17]: The Mean Absolute Error method is commonly used in age estimation task, which is shown in Eq. 1, where $\hat{y}$ and y are the prediction and the true label respectively and e_i is the absolute error,

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n}|\hat{y}_i - y_i|}{n} = \frac{\sum_{i=1}^{n}|e_i|}{n}. \tag{1}$$

- $\boldsymbol{\epsilon}$ **– error**: Another evaluation method is ϵ-error used in age estimation [17,58] to evaluate the algorithms performance. The standard formula is

$$\epsilon = 1 - \sum_{i=1}^{N} exp\left(-\frac{(\hat{y}_i - y_i)^2}{2\sigma_i^{*2}}\right), \tag{2}$$

where σ^* is the annotated standard deviation.

- **Multi-class ROC AUC** [4]: The shortage of the overall accuracy in the long-tailed visual problem is the decision threshold moved according to the training data distribution, which causes low prediction accuracy in balanced test sets. Calibrating the decision threshold may help with the overall accuracy. Thus, the receiver multi-class operating characteristic curve (ROC) and area under curve (AUC) could address the issue well. Specifically, ROC is a curve drawn with the x-axis showing the false positive rate (FPR), and y-axis the true positive rate (TPR) using different threshold values. AUC is the area under ROC curve representing the model performance by a value regardless of the threshold. One-vs-rest (ovr) and one-vs-one (ovo) are two strategies, meaning that computation AUC of each class against the rest or pairwise combination of classes. Multi-class ROC AUC is widely used in imbalanced learning and long-tailed learning to estimate classifier performance [4,63,64]. We choose multi-class ROC AUC ovr as the evaluation method in the experiment section.
- **Expected Calibration Error (ECE)**: Confidence calibration is a vital strategy to predict the probability of the true correctness likelihood [24,70,131]. The reliability diagrams are useful visual tools to get a scalar summary statistics of the calibration. It is indicated by the x-axis confidence and the y-axis accuracy, with the closeness to the diagonal indicating strong calibration performance. ECE is often used as a primary empirical metric to measure calibration by grouping N samples into M equal size bins. More specifically,

$$\mathrm{ECE} = \sum_{m=1}^{M}\frac{|B_m|}{N}|acc(B_m) - conf(B_m)|, \tag{3}$$

where the B_m represents the sample number falling into bin m, and acc and $conf$ are accuracy and confidence of B_m.

3.3. Empirical analysis

In this section, we first specify the problem setting and notations. Consider a visual recognition problem in the dataset $D = \{(x_i, y_i)\}, i \in \{1, 2, 3, \ldots, m\}$ with unknown distribution p, where x_i is the i^{th} image in the image set X and y_i is the corresponding label. $y_i \in Y = \{1, 2, 3, \ldots, C\}$ where C is the class number. Our goal is to learn a deep model $\mathbf{M} : X \to \mathbb{R}^{\mathbb{C}}$ by minimizing the misclassification error $p_{x,y}(y \neq \mathrm{argmax}\,\mathbf{M}(x))$. For the long-tailed setting, $p(y)$ exhibits lower-tail power-law distribution.

We describe the training process of deep models as follows. Input the training images to the DNN model $\mathbf{M}$ which could be represented by the feature extractor f and the classifier g. $\mathbf{x} = f(x, \Theta)$ and $\mathbf{z} = g(\mathbf{x})$, where $\mathbf{x} \in \mathbb{R}^D$ is the feature representation with dimension D, Θ is the parameters of extractor f and $\mathbf{z} \in \mathbb{R}^{\mathbb{C}}$ is the classifier logits $[z_1, z_2, \ldots, z_C]^\top$. $\hat{y} = argmax(\mathbf{z})$ is the class prediction. In most of the cases, $g(\mathbf{x}) = \boldsymbol{W}_c^\top \mathbf{x} + \boldsymbol{b}$ with the classifier weight matrix $\boldsymbol{W}_c \in \mathbb{R}^{D \times C}$ and the bias $\boldsymbol{b} \in \mathbb{R}^{\mathbb{C}}$. The probability calculated by softmax is represented as $p_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}}$. In long-tailed image recognition, the overall instance number $n = \sum_{j=0}^{C} n_j$ where n_j is the class frequency of class j. $U = \{\boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_C\}$ denotes the class centroids with the same dimension of $\mathbf{z}$.

Long-tailed training sets normally cause biased accuracy, meaning that the data-rich classes have higher accuracy than the data-poor classes. The conventional training process of DNNs is shown in Fig. 4. The mini-batch data is sampled from the training set by the instance balanced sampling strategy and then input to the deep model to get the logits. The cross-entropy loss is calculated from true labels and the softmax of logits as predictions. Following that, the DNN parameters are optimized by backpropagation with a

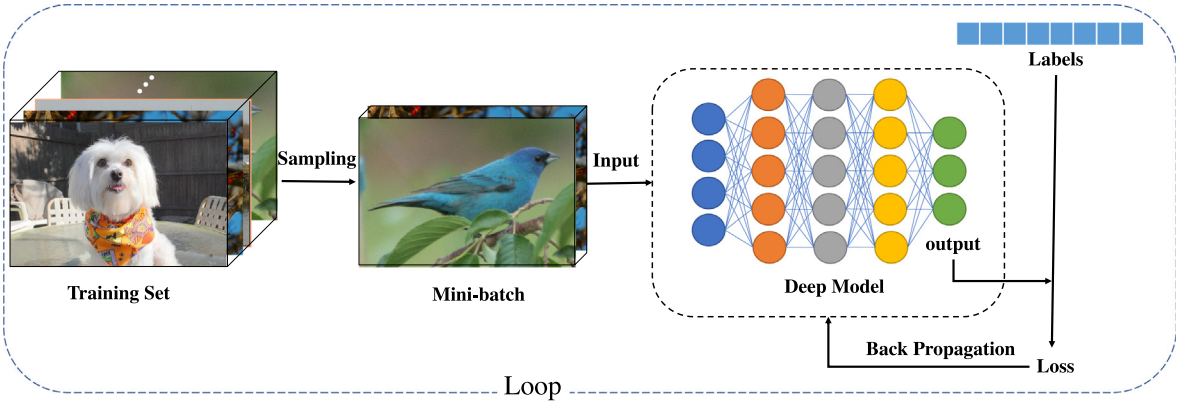

Fig. 4. DNN training process.

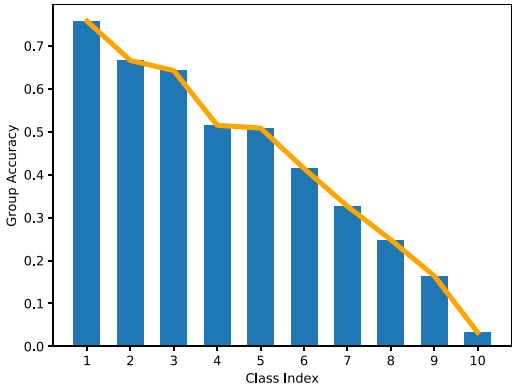

Fig. 5. Group Accuracy of ImageNet-LT dataset in the baseline model. 1,000 frequency-descending sorted classes are gathered into 10 groups. A considerable decreasing trend is discovered from head classes to the tail classes with high positive correlations to class frequencies.

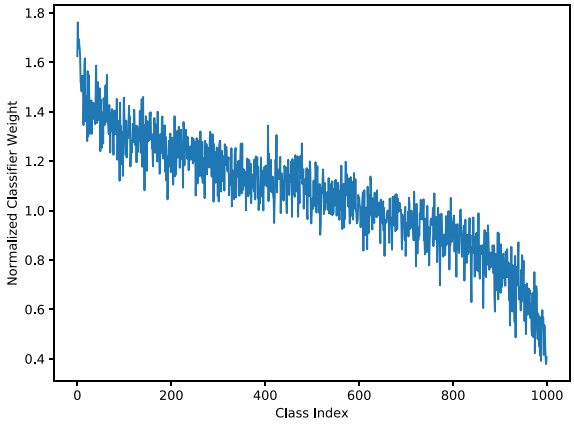

Fig. 6. Normalized classifier weights of ImageNet-LT dataset in the baseline model. Labels indicate the class index in a long-tailed distribution. Clearly, a positive correlation of the normalized classifier weights and the class frequencies is shown.

given learning rate, schedular and momentum. This process will repeat for given epochs to get the optimal outputs. In the long-tailed visual problems, training data, sampling strategy, model structure, loss function, optimization strategies and test data distribution are all responsible for the poor performance of deep models. To begin with, the highly imbalanced distribution and the under-representativeness of low-shot categories are the primary issues. Secondly, the random sampling method, which treats each instance equally, generates imbalanced mini-batch data. Then, model structure, loss function and optimization strategies ignore the highly imbalanced input and cause the final results dominated by head classes. Finally, the performance of DNN is unpredictable facing the unknown test distributions.

To illustrate the model and performance impact from the long-tailed data, the group accuracy and the normalized classifier weights are shown in Figs. 5 and 6. Fig. 5 shows the group accuracy of ImageNet-LT dataset in ResNeXt-50. 1,000 frequency-descending sorted classes are gathered into 10 groups. The group accuracy is the mean accuracy of the group classes. There is a considerable decreasing trend both in group accuracy and normalized classifier weights, which indicates the positive correlation with the class frequencies. The imbalanced classifier weights could be the result of biased accuracy.

The confusion matrix of CIFAR 100 im100 dataset is presented in Fig. 7. The x and y axes are the predicted labels and true labels, respectively. The colour of each pixel indicates the instance numbers from the true label to the predicted label. In this matrix, the diagonal means the instance numbers with correct predictions. This line exhibits a gradual fading of brightness from the top left corner to the bottom right corner, indicating a decrease in class accuracy from data-rich classes to data-poor classes. Meanwhile, light pixels in the lower-left region illustrate that the minorities have higher probabilities of being misclassified to the majority classes.

4. Methods

In this section, the algorithms addressing the long-tailed visual recognition problem are divided into three categories namely: training stage, fine-tuning stage, and inference stage. Fig. 8 shows the trend in three categories over the last five years (2017–2021). Although, the numbers of publications increase in fine-tuning and inference stage methods, more recent research focus has been shifted to the multiple experts and transfer learning strategies.

4.1. Training stage

Algorithms in training stage include data augmentation [8,9,49,99,103,110,123,128], re-sampling [4,6,13,40,53,68,69], cost-sensitive loss [12,17,33,41,51,66,73,77–79,85,88,93,104,115,127,131], and multi-expert and transfer learning methods [5,10,13,19,26,36,37,50,54,56,60,84,105,107,109,117,118,130,132,135]. We will introduce some of the popular strategies in this part. The number of publications with respect to the different strategies

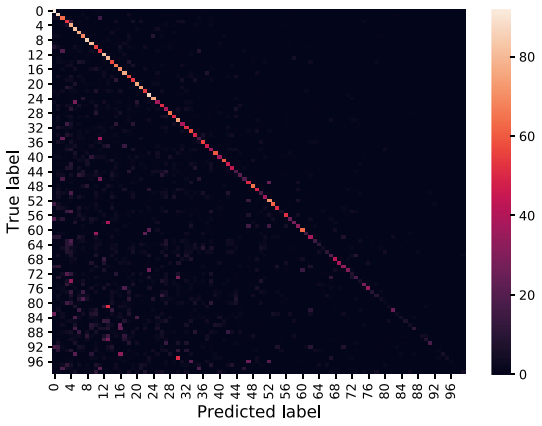

Fig. 7. The confusion matrix of CIFAR-100 im100 dataset. Labels indicate class indexes in a long-tailed distribution. Prediction accuracy in head classes (upper left) is higher (lighter) than in the tail (bottom right). Moreover, the bottom left section has light spots that indicate instances have a high misclassification probability to data-rich classes.

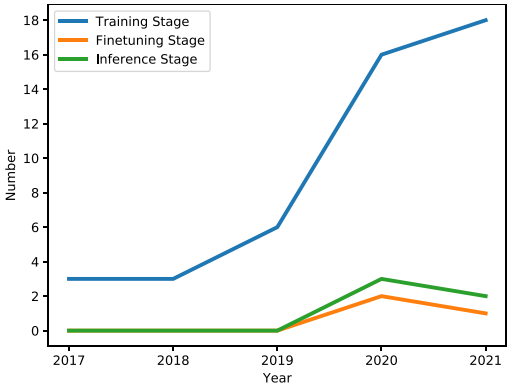

Fig. 8. Number of publications in training, fine-tuning and inference stage methods.

in this stage over the last five years is depicted in Fig. 9, which illustrates that cost-sensitive loss, multiple experts and transfer learning based methods got intensive attention, while re-sampling strategies got the least.

4.1.1. Data augmentation

Data augmentation solutions including mixup based methods, generative adversarial networks (GANs) and semi-supervised learning methods are developed to address the paucity of instances

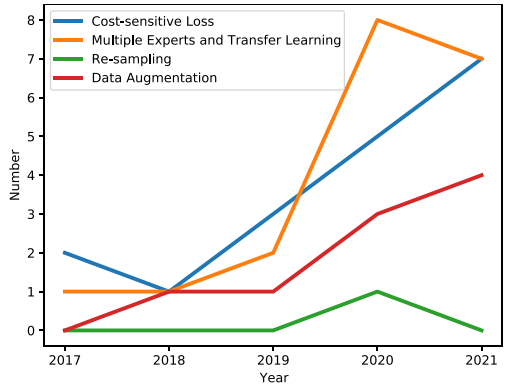

Fig. 9. Number of publications in training stage methods including cost-sensitive loss, multiple experts and transfer learning, re-sampling and data augmentation.

and representativeness. These methods aim to enhance the feature representations of under-represented classes at image and feature levels.

Mixup based methods [8,99,119,123] enrich the training data with linear interpolation of randomly selected instances. The mixup interpolation method is described as: $\tilde{x} = \lambda x_i + (1-\lambda)x_j, \tilde{y} = \lambda y_i + (1-\lambda)y_j$, where x_i, x_j are raw input vectors; y_i, y_j are one-hot label encoding; $\tilde{x}$ and $\tilde{y}$ are the generated image and label; λ is a hyper-parameter. Then manifold mixup [99] adopted the same interpolation strategy as the mixup in the hidden feature space of neural networks. Experiments show that the manifold mixup is beneficial for a more flatten and clearer decision boundary. While mixup and manifold mixup are general augmentation methods, there are also mixup-based methods particularly designed for long-tailed recognition task. Remix [8] was proposed to adjust the interpolation coefficient λ according to the head or tail class the current training sample belongs to. MixSMOTE [119] assigned higher probabilities for tail classes to be selected when mixing up. The core issue of these mixup based methods is to design proper data selection and interpolation strategy so that the biased decision boundary could be rectified.

Meanwhile, M2m [43], ImbalanceCycleGAN [83], MetaS-Aug [49] and CAM-based sampling [128] attempt to augment the minority classes with more diversity. M2m learns to augment the minority classes by translating the majority class samples via an adversarial-like training. The ImbalanceCycleGAN is based on CycleGAN [134] to generate tail class samples from the head to expand the feature space for minorities. This method is proved to be effective in three datasets including CelebA [59], CUB-200–2011 [102] and Horse2Zebra [134]. MetaSAug learns class-wise covariance matrices containing semantic directions and augmenting minority classes. The covariance matrices are treated as trainable parameters and are updated from the validation set to get the optimal value. Then, it is used to optimize the classifier. The CAM-based sampling is an augmentation strategy that transforms the foreground and keeps the background unchanged to generate a series of new images. The separation of foreground and background is based on the Class Activation Map (CAM) [133] values.

Another intuitive idea for long-tailed data distribution is to directly increase training instances in data-poor classes from related datasets. Semi-supervised learning could generate pseudo labels for unlabeled data to reduce sample scarcity in tail classes and balance the training sets. To this end, semi-supervised learning for long-tailed recognition problems has also been introduced and achieved promising results [55,110,121]. Yang and Xu [121] first proposed that the extra data in semi-supervised method could reduce label bias and improve the final classifier. Following this idea, Liu et al. [55] found out that the pseudo label performs poorly on tail classes and can hardly be leveraged. To address this issue, a framework that combined the model decoupling method and class-balanced sampling strategy was proposed to generate more accurate pseudo labels. Similarly, to tackle the semi-supervised bias, Wei et al. [110] introduced self-training with class re-balancing to train the model for multiple generations. These methods provide novel ideas for long-tailed data re-balancing.

In summary, this subsection discussed the training stage methods based on data augmentation. The mixup based methods are usually easy to implement with just a few lines of code. However, since mixup and its variants are generic augmentation methods, a particular design for long-tailed distribution is needed. Although various augmentation methods aim to augment the tail class samples, they usually improve the performance at the cost of extra computation (e.g., GAN network, adversarial training, CAM). The key to these methods is how to efficiently generate realistic and diverse minority class samples. Finally, semi-supervised learning

methods demonstrate a practical path towards achieving performance gain. However, they may also be limited by the relevance between the target long-tailed dataset and the auxiliary datasets.

4.1.2. Re-sampling strategy

Stochastic gradient descent (SGD) is effective in parameter optimization of deep models. To overcome the time-consuming of one-example-each-iteration strategy, the mini-batch SGD optimization was proposed which randomly samples a batch of data from the dataset. It effectively balances the accuracy and training time with smooth convergence. The random sampling (instance-balanced) method, where each instance has an equal probability to be selected is shown as

$$p_j^I = \frac{n_j}{\sum_{i=1}^{C} n_i}, \tag{4}$$

where p_j^I is the selective probability of instance in class j, C is the number of classes in the dataset [40]. In long-tailed visual tasks, the instance balanced sampling method causes the imbalanced performance in each mini-batch. After summing up the cross-entropy loss of the imbalanced batch data, the head categories would dominate the gradient descent orientations of parameters. Therefore, the deep models are biased to the data-rich classes and show poor performance to the data-scarcity categories.

A number of re-sampling procedures, such as class-balanced sampling and square-root sampling approaches, are introduced in response to the aforementioned problem. They adjust the instance sampling probability to reduce the mini-batch distribution's imbalance level. We will go through a few simple but effective re-sampling methods in this section, including random over-sampling, random under-sampling, class-balanced sampling, square-root sampling, and progressively balanced sampling methods.

- **Random Over-sampling**: One simple way for balancing the training batch is to use a random over-sampling strategy that stochastically repeats a few categories several times while maintaining the number of head classes [53]. The instance sampling probability is

$$p_j^I = \frac{\lambda n_j^q}{\sum_{i=1}^{C} n_i^q}, \tag{5}$$

  where the value of q is equal to 1 for head classes while for tail classes, $q \in (0, 1)$. The hyperparameter $\lambda = 1$. This method repeats the instance without adding new knowledge in training models and maintains the amount of data compared with the instance balanced sampling strategy. The over-sampling method is likely to cause over-fitting to the classic machine learning models, but when training deep convolutional networks, experiments show that the classifier could learn the decision boundary well without over-fitting [4].
- **Random Under-sampling**: In contrast with the random over-sampling method, the random under-sampling method removes the instances unselected from data-rich categories to even the mini-batch distribution. In Eq. 5, $q \in (0, 1)$ for head classes and $q = 1$ for tail classes. The hyperparameter $\lambda = 1$. A straightforward detriment is discarding a fraction of training data, which might leave out key features and cause relatively poor feature representation.
- **Class-balanced Sampling**: Without focusing on the head or tail class frequencies, class-balanced sampling treats each class equally with the same probability to be chosen. The probability of sampling an instance from class j, p_j^{CB} is equal to 1/C, that is in Eq. 5, $q = 0, \lambda = 1/n_j$. The sampling could be separated into two stages, where the first stage is to sample a class with uniform probability, and the second is to randomly select an instance from that class [40]. In that strategy, the under-sampling and the over-sampling might co-exist.
- **Square-root Sampling**: The high imbalance of long-tailed distribution results in the low performance in the minorities. Experiments show that the lower imbalanced ratio is beneficial to the final performance when the training number of instances is fixed [4]. Therefore, the square-root sampling method is proposed to alleviate the imbalanced ratio where the $q = 1/2, \lambda = 1$ in Eq. 5 [62,68].
- **Progressively Balanced Sampling**: Despite the consistent re-sampling methods during the training process, the progressively balanced (PB) sampling method [13] and deferred re-sampling [6] change the sampling probability gradually from instance balanced (IB) to class balanced (CB) strategy, where the probability adjusts in every epoch. It could be illustrated as Eq. 6, where t is the current training epoch, and T is the total training epoch.

$$p_j^{PB}(t) = \left(1 - \frac{t}{T}\right) p_j^{IB} + \frac{t}{T} p_j^{CB} \tag{6}$$

  The purpose of this strategy is to transfer the knowledge and get a re-balanced network gradually. Experiments show that this is beneficial in overall performance, especially for the tail classes [13]. Different values of q in Eq. 5 with brief re-sampling descriptions are concluded in Table 2.

Apart from manually designing sampling rules, there are also methods to automatically learn a sampling strategy. Peng et al. [74] proposed a trainable undersampling method for imbalance classification. They parameterized the data sampler and used reinforcement learning for optimization. Ren et al. [78] used meta learning to find an optimal sampling rate for each class.

In summary, the re-sampling methods mainly focus on adjusting the sampling probability. The basic rule of hand-crafted re-sampling methods is to assign larger probabilities for the minority classes. The automatic re-sampling methods, on the other hand, resort to techniques such as meta-learning or reinforcement learning for optimization. The common practice of them is to parameterize the sampling process such that the sampling strategy can be optimized. Re-sampling methods are generally beneficial for the long-tailed performance, especially in the extreme imbalanced

Table 2
Different q and λ in Eq. 5 with briefly re-sampling descriptions.

Re-sampling methods	Value of q and λ	Description
Instance balanced sampling	$q = 1$, $\lambda = 1$	Each instance has the same probability of being chosen.
Random Over-sampling [4]	*Head* : $q = 1$ *Tail* : $q \in (0, 1)$ $\lambda = 1$	Randomly over-sampling the tail classes and maintaining the instance balanced sampling to head classes.
Random Under-sampling [69]	*Head* : $q \in (0, 1)$ *Tail* : $q = 1$, $\lambda = 1$	Randomly under-sampling the head classes and maintaining the instance balanced sampling to tail classes.
Class-balanced Sampling [40]	$q = 0$, $\lambda = 1/n_j$	Each class has the same probability of 1/C.
Square-root Sampling [62,69]	$q = 0.5$, $\lambda = 1$	Relatively balanced than the long-tailed distribution.
Progressively Balanced Sampling [13,6]	$q = 1$ to $q = 0$, $\lambda = 1$ to $\lambda = 1/n_j$	Changes the sampling probability gradually from instance balanced to class balanced sampling method.

cases (e.g., minority classes only have less than 10 samples). However, since the tail class instances are frequently oversampled, the models are at the risk of overfitting tail classes.

4.1.3. Cost-sensitive loss

Loss controls the optimisation direction of the parameters throughout the training phase. The most prevalent mechanism in the training process, the cross-entropy (CE) loss and the SGD optimisation method, treat each instance identically, resulting in the overlook of the tail classes. The CE loss is represented by Eq. 7, given the prediction outputs $\mathbf{z} = [z_1, z_2, \ldots, z_C]^\top$,

$$L_{\mathrm{CE}} = -\log \frac{e^{z_c}}{\sum_{i=1}^{C} e^{z_i}}. \tag{7}$$

The objective of the cost-sensitive loss is to improve overall performance, particularly the accuracy of tail classes in long-tailed visual tasks, by implementing better punishment strategies, such as adding weight and enlarging tail class margins in loss functions.

For imbalanced datasets, assigning weights by inverse or square-root inverse of the class frequencies [34,62,68,109] are generally adopted. Focusing on large-scale, highly imbalanced data, these strategies cannot yield satisfactory results. Cui et al. [12] considered the effective number of samples to address the information overlap in data. Then, class-balanced term is adopted to smooth the balanced strategy which could be represented by $(1-\beta)/(1-\beta^{n_c})$ where the β is a hyperparameter. This term is applicable to other loss functions.

Following this idea, Jamal et al. [37] improved the re-weighting method by introducing a conditional weight learned from a meta-learning framework. This method is intuitive while time-consuming compared to other tricks.

To avoid the discouraging gradient from majority classes, the softmax equalization loss (SEQL) [93] introduced class frequencies with abandon function in softmax loss to disregard a proportion of discouraging gradient as

$$\tilde{p}_i = \frac{e^{z_i}}{\sum_{k=1}^{C} \tilde{w}_k e^{z_k}}, \tag{8}$$

and $\tilde{w}_k$ is calculated by

$$\tilde{w}_k = 1 - \hat{\beta} T_\lambda \left(\frac{n_k}{\sum_{j=1}^{C} n_j} \right) (1 - y_k), \tag{9}$$

where $n_k / \sum_{j=1}^{C} n_j$ is the frequency of class k and $\hat{\beta}$ is a random variable with the probability γ and $1-\gamma$ to be 0 and 1. T_λ is a threshold function with threshold λ.

Seesaw loss [104] combines the class frequencies and the logits to generate a balanced loss shown as

$$L_{\mathrm{seesaw}}(\mathbf{z}) = -\sum_{i=1}^{C} y_i \log\left(\hat{\sigma_i}\right) \quad \text{with } \hat{\sigma_i} = \frac{e^{z_i}}{\sum_{j \neq i}^{C} \mathbf{S}_{ij} e^{z_j} + e^{z_i}}, \tag{10}$$

$$\mathbf{S}_{ij} = \mathbf{M}_{ij} \cdot \mathbf{C}_{ij}, \tag{11}$$

$$\mathbf{M}_{ij} = \begin{cases} 1, & \text{if } N_i \leqslant N_j \\ \left(\frac{N_j}{N_i}\right)^p, & \text{if } N_i > N_j \end{cases}, \tag{12}$$

$$\mathbf{C}_{ij} = \begin{cases} 1, & \text{if } \sigma_j \leqslant \sigma_i \\ \left(\frac{\sigma_j}{\sigma_i}\right)^q, & \text{if } \sigma_j > \sigma_i \end{cases}. \tag{13}$$

The $\mathbf{M}_{ij}$ and $\mathbf{C}_{ij}$ represent the mitigation factor and the compensation factor. The former alleviates the tail class penalty according to the class frequency ratio of the tail class i and the head class j. The compensation factor enlarges the penalty based on the mismatch of class i. q and σ_i are the hyper-parameter and the predicted probability of class i.

Without considering the class frequency directly, focal loss [51] noticed the relationship between the tail classes and the prediction difficulties and introduced a modulating factor corresponding to the instance difficulties (logits values) to cross-entropy loss. It is defined as:

$$L_{\mathrm{Focal}} = -(1-p_t)^\gamma \log(p_t), \tag{14}$$

where the p_t is the probability of class t defined as:

$$p_t = \begin{cases} p_t, & t = c \\ 1 - p_t, & t \neq c \end{cases}. \tag{15}$$

The hyperparameter γ controls the sample importance.

Similarly, taking uncertainty into consideration, DRO-LT [85] was proposed based on robustness theory. It pushes and pulls the estimation error towards a worst-case possible distribution, resulting in larger uncertainty areas for tail classes compared with the data-rich categories. It is denoted as

$$L = \lambda L_{CE} + (1-\lambda) L_{\mathrm{Robust}}, \tag{16}$$

where

$$L_{\mathrm{Robust}} = -\sum_{c \in C} w(c) \sum_{\mathbf{z} \in D_c} \log \frac{e^{-d\left(\widehat{\boldsymbol{\mu}_c}, \mathbf{z}\right) - 2\varepsilon_c}}{\sum_{\mathbf{z}'} e^{-d\left(\widehat{\boldsymbol{\mu}_c}, \mathbf{z}'\right) - 2\varepsilon_c \delta(\mathbf{z}', c)}}. \tag{17}$$

$\hat{\mu_c}$ represents the empirical centroid of class $c, \delta(\boldsymbol{z}, c) = 1$ if $\mathbf{z}$ is of the class c and 0 otherwise. $\varepsilon_c = \sigma_c \sqrt{2} \epsilon_c$ where σ is the radius of the uncertainty set U_c and ε_c is the variance of the distribution for a sample in class c. The uncertainty set could be written as $U_c := \{q | D_{KL}(q \| \widehat{p_c}) \leqslant \epsilon_c\}$, where $D_{KL}(q \| \widehat{p_c}) = \frac{1}{2\sigma^2} d\left(\boldsymbol{\mu}_q, \boldsymbol{\mu}_{\widehat{p_c}}\right)^2$. d is the Euclidean distance.

Cao et al. [6] focused on the margin of tail classes and proposed a label-distribution-aware margin loss (LDAM) enlarging the margin of the minority classes. By adding a parameter related to the class frequencies in vanilla cross-entropy loss, the tail classes attempt to have larger margins to other classes, which helps the model adapts to the evenly distributed test set. The LDAM loss is represented by:

$$\begin{aligned} L_{\mathrm{LDAM}} &= -\log \frac{e^{z_c - \Delta_c}}{e^{z_c - \Delta_c} + \sum_{j \neq c} e^{z_j}}, \\ \text{where } \Delta_j &= \frac{K}{n_j^{1/4}} \quad \text{for } j \in \{1, \ldots, C\}, \end{aligned} \tag{18}$$

and K is a constant to be defined.

The progressive margin loss was introduced for the age estimation task by merging an ordinal margin learning and a variational margin learning [17]. Class centre, intra-class variance, and inter-class variance are the three variables used to determine these margins. To evaluate the distribution similarity, the loss is trained using the Kullback Leibler (KL) divergence. In comparison to state-of-the-art approaches, it achieved compelling performance in the age estimation task.

Focusing on disentangling the source label distribution from the model prediction, Hong et al. [33] proposed post-compensated softmax (PC Softmax) and label distribution disentangling loss

(LADE). The former directly disentangling the source label distribution from the model prediction in training while the latter is based on the optimal bound of Donsker-Varadhan representation. The disentangling losses are useful tools in long-tailed visual recognition and also effective in terms of confidence calibration.

Cui et al. [11] explored the effective supervised contrastive loss for the image representation learning phase and proposed parametric contrastive learning (PaCo). The basic idea of the contrastive learning is to pull the instances in the same category together and push the instances in different classes apart. Instead, they introduced a set of learnable category centers and balanced PaCo loss to pull the samples together with centers and benefit hard example learning. The PaCo loss can be presented as

$$L_i = \sum_{\mathbf{z}_+ \in p(i) \cup \{\mu_y\}} -w(\mathbf{z}_+) \log \frac{\exp(\mathbf{z}_+ \cdot T(x_i))}{\sum_{\mathbf{z}_k \in A(i) \cup \mathbf{U}} \exp(\mathbf{z}_k \cdot T(x_i))}, \tag{19}$$

where $p(i) = \{z_k \in A(i) : y_k = y_i\}, A(i) = \{z_k \in queue \cup Z_{v1} \cup Z_{v2}\} \setminus \{z_k \in Z_{v1} : k = i\}$.. Z_{v1} and Z_{v2} are outputs from query and key networks, μ_y is the learnable center of class y,

$$w(\mathbf{z}_+) = \begin{cases} \alpha, & \mathbf{z}_+ \in P(i) \\ 1.0, & \mathbf{z}_+ \in U \end{cases}, \tag{20}$$

α is a hyper-parameter in $(0, 1)$. In the paper, it is set to 0.05.

$$\mathbf{z} \cdot T(x_i) = \begin{cases} \mathbf{z} \cdot \mathscr{G}(x_i), & \mathbf{z} \in A(i) \\ \mathbf{z} \cdot \mathscr{F}(x_i), & \mathbf{z} \in U, \end{cases} \tag{21}$$

where U is the center set, the transform $\mathscr{G}$ and $\mathscr{F}$ are two-layer MLP and identity mapping $\mathscr{F} = x$ relatively. The model is trained using PaCo loss for 400 epochs to get better results, since contrastive learning has a higher computational cost compared to CE related losses.

Similarly, inspired by contrastive loss, Zhang et al. [127] presented range loss increasing inter-class distance and shrinking the intra-class distance formulated as $L_R = \alpha L_{R_{\text{intra}}} + \beta L_{R_{\text{inter}}}$, where

$$L_{R_{intra}} = \sum_{i \subseteq I} L^i_{R_{intra}} = \sum_{i \subseteq I} \frac{k}{\sum_{j=1}^{k} \frac{1}{D_j}}, \tag{22}$$

$$\begin{aligned} L_{R_{\text{inter}}} &= \max(M - D_{\text{Center}}, 0) \\ &= \max\left(M - \|\bar{\boldsymbol{x}}_Q - \bar{\boldsymbol{x}}_R\|_2^2, 0\right). \end{aligned} \tag{23}$$

D_j is the largest distance among the features in one class, $\bar{x}$ is the class center, Q and R are two nearest classes among the mini-batch and M is a hyper-parameter representing the max optimization margin. Consequently, the range loss compresses the intra-class feature space and enlarges the intra-class distance by maximizing the center distances.

In the multi-label task with more than one ground-truth label, the traditional loss is binary cross-entropy (BCE) [115] as

$$L_{BCE}(\mathbf{z}) = -\frac{1}{N} \sum_{i=1}^{N} y_i \cdot log(z_i) + (1 - y_i) \cdot \log(1 - z_i). \tag{24}$$

To counter the challenges in multi-label classification, arisen by the co-occurrence of labels and the dominating effect caused by negative labels, the distribution-balanced loss is proposed and is given as:

$$\begin{aligned} L_{DB}(\mathbf{z}, y) \quad &= \tfrac{1}{C} \sum_{i=0}^{C} \hat{r}_i \left[y_i \log\left(1 + e^{-(z_i - v_i)}\right) \right. \\ &\left. + \tfrac{1}{\lambda}(1 - y_i) \log\left(1 + e^{\lambda(z_i - v_i)}\right) \right], \end{aligned} \tag{25}$$

where λ, v_i and $\hat{r}_i$ are scale factor, class-aware bias and re-weighting factor. It combines two strategies which are re-balanced weights and negative tolerant regularization. The first strategy helps to balance category sampling probability based on the multi-label scenario as given in Eq. 26 indicating a smooth function based on r_i, where the weight r_i is shown by Eq. 27. β and μ are hyperparameters controlling the shape of the mapping function. The p_i^C and p^I are class level probability and instance-level probability shown as Eq. 28.

$$\hat{r}_i = \alpha + \frac{1}{1 + \exp(-\beta \times (r_i - \mu))}, \tag{26}$$

$$r_i = \frac{p_i^C(x)}{p^I(x)}, \tag{27}$$

$$p_i^C(x) = \frac{1}{C} \frac{1}{n_i}, p^I(x) = \frac{1}{C} \sum_{y_i=1} \frac{1}{n_i}. \tag{28}$$

In summary, cost-sensitive methods usually calibrate cross-entropy loss bias by assigning different weights based on the class frequencies, sample difficulty, distances to centroids or class margins. These methods are usually lightweight and can be easily plugged into various frameworks or architectures. However, cost-sensitive methods are limited in the extreme imbalanced case. In this case, the tail class instances can hardly be seen during training without integrating with other techniques such as re-sampling so that the cost-sensitive loss becomes ineffective. For instance, the LDAM [6] loss has to be combined with deferred re-sampling to achieve better performance. Thus, investigations of cost-sensitive methods that can be effectively incorporated with other techniques are needed.

4.1.4. Multiple experts and transfer learning

The considerable difference between head classes and tail classes contributes to the bias in feature learning and the classifier. Therefore, many researchers have endeavored to transfer the knowledge from data-rich categories to few-shot classes by distilling from multiple experts or specially designed branches. Knowledge learned from long-tailed datasets is divergent in different sampling strategies, which is exploited by many methods to improve the few-shot feature representation.

Probability threshold bagging (PT-bagging) [10] combines the bagging ensemble and threshold-moving strategy based on decoupling the training of the encoder and the classifier separately. The ensemble method means combining over-sampling and under-sampling methods to sample the training data into m groups and then learns a series of classifiers. When testing, the averages of each class predictions are calculated from the m inference logits with the input image. Then, the final logits are obtained by $p(y = k|x)/p(y = k)$ where the $p(y = k)$ could be estimated from the class frequency ratio as $n_k / \sum_{j=1}^{C} n_j$. This method is transferable to many tasks including multi-class tasks. Besides, the moving-threshold method can also be applied as a post-hoc strategy.

Similar to PT bagging that transfers knowledge among different sampling methods, GistNet [54] combined the random sampling loss and the class-balanced sampling loss with a given ratio. It attempts to exploit the overfitting caused in head classes and transfer the geometry from head to tail. This is profitable for the few-shot generalization without class weight specification. Likewise, [26] designed a network with two branches separately trained from uniform sampling method and re-balanced sampling method. Then a binary-cross-entropy-based classification loss was defined to learn the consistency between branches. During the test phase, the cross-branch paths are ignored to get the average predictions from two subnets. This method is also applicable to the multi-label visual recognition task.

Zhang et al. [124] introduced auxiliary learning, which combines class-balanced sampling classifier h, instance balanced sampling

classifier h_a and self-supervised learning classifier h_s [22]. The self-supervised learning task applies random rotation to original images and is trained to learn the rotation angle. In the training process, the classifiers are solely optimized by each different learning strategy, but the features are trained using the combination of different classifier losses. The final loss is $L_{\text{Final}} = \lambda_1 L_{\text{CBS}} + \lambda_2 L_{\text{RRS}} + \lambda_3 L_{\text{SS}}$. λ_1, λ_2 and λ_3 are three hyperparameters. The feature extractor and the class-balanced classifier are retrained for prediction.

Zhou et al. [132] discovered that re-balancing strategies (re-sampling and re-weighting) effectively improve the performance in the evenly distributed test set, while the drawback is the impairment to the feature generalization. Therefore, BBN is proposed to take care of the representation learning and the classifier simultaneously. The architecture is constructed by two branches trained by different sampling strategies: instance sampling strategy and reversed sampling method, which shares three residual blocks (trained by ResNet or ResNeXt Networks). At the junction of both sampling method models, an "Adaptor" parameter is enabled to automatically balance the output by both methodologies and produce a tail-distribution adapted learner. In the CIFAR-10-LT dataset, the proposed parabolic decay has the lowest error rate when compared to other adaptive strategies. In the inference phase, the adaptor is fixed to 0.5 to treat each branch equally.

To further explore the multi-stage training scheme, [50] proposed SSD method that combined the self-supervised learning and the multi-stage training with different sampling methods. Self-supervised loss is introduced to provide less biased sorted labels for the following stages. Specifically, there are three steps for SSD, which are feature learning stage guided by self-supervision, soft labels generation by tuning classifier and joint training with self-distillation.

The high imbalance ratio in long-tailed visual tasks might be the main cause of poor tail class performance. Therefore, Xiang et al. [118] proposed the Learning From Multiple Experts (LFME) that attempted to alleviate the imbalance level by dividing the cardinality-sorted training set into parts, where $D = S_0, S_1, \ldots, S_L, S$ is the cardinality-adjacent subset separated by $L-1$ thresholds. Then, each group is trained in an expert model. There is a higher accuracy when comparing the expert and the model trained from the whole dataset in the corresponding group. Next, a scheme of learning a student model from multiple experts is proposed. It is constructed with two elements, self-paced expert selection and curriculum instance selection. In the former, the assigned weights are determined by looking at the accuracy difference between the student model and the experts, and the latter learns the instances from easy to hard based on the prediction probabilities. The training loss for the student model is

$$L = \sum_{i=1}^{N} v_i^k L_{CE}(x_i, y_i) + \sum_{l=1}^{L}\sum_{i=1}^{N} w_l L_{KD_l}(\boldsymbol{M}, \boldsymbol{M}_{Exp}; x_i), \tag{29}$$

where L_{CE} and L_{KD} are cross entropy loss and knowledge distillation loss. $\boldsymbol{M}$ and $\boldsymbol{M}_{Exp}$ are student model and expert models accordingly.

$$w_l = \begin{cases} 1.0 & \text{if } Acc_M \leqslant \alpha Acc_{E_l} \\ \frac{Acc_{E_l} - Acc_M}{Acc_{E_l}(1-\alpha)} & \text{if } Acc_M > \alpha Acc_{E_l} \end{cases}, \tag{30}$$

where Acc_M and Acc_{E_l} are the accuracy of student model and expert models, α is the expert weight scheduling threshold.

$$v_i^k = \left(1 - v_i^{(1)}\right)\frac{e}{E} + v_i^{(1)}, v_i^{(1)} = p_i \frac{N_{S_{min}}}{N_{S_l}}. \tag{31}$$

v_i^k will gradually grow from $v_i^{(1)}$ to 1. e and E are the current epoch and the total epoch number. $N_{S_{min}}$ and N_{S_l} are minimum cardinality and class l cardinality in subset S. It proved that an ensemble of models could increase the performance of a single model.

Unlike LFME that distills knowledge from shot-based teacher models, class-balanced distillation (CBD) [36] trains teacher models with different data augmentations. Specifically, the standard model and data augmentation models are presented, which are trained by random crop and flip as well as color jitter and Gaussian noise. Moreover, they started by using different initial random seeds to affect initialization and the order of classes in the training process. In the distilling stage, the student model is encouraged to heed the feature extraction from the teacher models. The results show that feature level distillation works better than classifier and hybrid distillation strategies. And learning from four teacher models with equal or similar numbers of vanilla models and data augmentation models can get improved results.

Model bias is an inherent problem in long-tailed vision tasks. Routing Diverse Experts (RIDE) [107] method was proposed to integrate diverse expert models with different feature distributions by expert assigning models. It is trained in two stages. Firstly, k expert models that share top weights (two residual blocks taking ResNet as an example) are trained through the proposed distribution-aware diversity loss and classify loss. The classified loss could be LDAM loss, focal loss or other effective losses. The total loss is

$$\begin{aligned} L_{\text{Total}}^i &= L_{\text{Classify}}^i\left(\phi^i(\boldsymbol{x}), y\right) \\ &- \tfrac{\lambda}{n-1}\sum_{j\neq i}^{n} D_{\text{KL}}\left(\phi^i\left(\boldsymbol{x}, \vec{T}\right), \phi^j\left(\boldsymbol{x}, \vec{T}\right)\right), \end{aligned} \tag{32}$$

where D_{KL} is the KL divergence loss that encourages diversity in minorities, $\vec{T}$ is the temperature vector assigning each instance a single temperature T, which calculated as

$$\begin{aligned} T_i &= \eta\psi_i + \eta(1 - max(\Psi)), \\ \Psi = \{\psi_1, \ldots, \psi_C\} &= \left\{ \gamma \cdot C \cdot \frac{n_i}{\sum_{k=1}^{C} n_k} + (1-\gamma) \right\}_{i=1}^{C}. \end{aligned} \tag{33}$$

Then, the expert assigning models are trained with fixed expert weights and routing loss:

$$\begin{aligned} L_{\text{Routing}} &= -\omega_{\text{p}} y \log\left(\tfrac{1}{1+e^{-y_{\text{ea}}}}\right) \\ &-\omega_{\text{n}}(1-y)\log\left(1 - \tfrac{1}{1+e^{-y_{\text{ea}}}}\right), \end{aligned} \tag{34}$$

where $y_{\text{ea}} = \mathbf{W}_2(\mathbf{l}_i \oplus \sigma(\mathbf{W}_1\mathbf{x}_i))$. The $\oplus$ is concatenation, $\sigma(\cdot)$ is the RELU function, $\mathbf{l}_i$ and $\mathbf{x}_i$ are top k logits and feature in expert i. In the training process, if the current expert gets the wrong prediction while the following expert makes a correct prediction, y is set to 1. In other cases, it is set to 0. In this way, the routing models are encouraged to assign hard samples to other experts with different feature distributions. The experiments show that increasing the number of experts (from 1 to 8) could improve overall accuracy, particularly for the tail classes.

Wang et al. [105] tackled the high memory cost in training large datasets such as iNaturalist and presented standard supervised contrastive (SC) loss and prototype supervised contrastive (PSC) strategies to address long-tailed problems. The prototype could be treated as centers of individual classes when compared with PaCo. However, in the training process, Wang et al. [105] adopted two branches for feature learning and classifier learning. The losses are contrastive loss and CE loss respectively with an adaptive rate α. The linear decayed weighting factor α controls the weights

between the two losses and gradually transfers the attention from feature learning to classifier learning.

In summary, multiple experts and transfer learning methods focus on exploiting the external or ensemble knowledge to enhance the model performance. The key insight is that models with different training configurations (e.g., with different sampling techniques, or on different subsets) could be made complementary to each other, thus improving the overall performance. However, the drawback of these methods is also obvious. They usually require large computational costs with complicated architectures or training procedures.

In this section, methods in the training stage are revisited including data augmentation, re-sampling methods, cost-sensitive loss as well as multiple experts and transfer learning. They cover the whole training process from data preparation, sampling strategy, model structure, loss function and training process, which are the mainstream approaches in the long-tailed recognition problem.

4.2. Fine-tuning stage

Methods in fine-tuning stage [39,40,57] first train feature representation in conventional strategy including uniform re-sampling method and CE loss and then fine-tune the classifier with different strategies.

Kang et al. [40] proposed an simple yet surprisingly effective method where the representation learning and the classifier training are decoupled. They observed that the representation learning network (feature learning network) is more generalizable when utilizing the randomly sampling method. Re-balancing, on the other hand, hampers the representation generalization. Then they proposed to train the whole network with randomly sampling in the first stage, and fine-tune the classifier with balanced sampling while keeping the representation network fixed. Experiments show that this simple method improves the proficiency of tail classes significantly and achieves promising results on several benchmark datasets.

Moreover, Zhang et al. [126] argued that there is still room for improvement between the decoupled training and the upper bound of the classifier. They first jointly train backbone models in the ImageNet-LT dataset and fine-tune the classifier in the original balanced ImageNet dataset. The result shows that there are still gaps between the long-tailed re-trained classifier and the original balanced re-trained classifier. Then they proposed DisAlign which first introduced an adaptive calibration function and then the confidence-aware distribution alignment employs a generalized re-weighting scheme for balanced class distribution. In the first stage, the adaptive calibration function is

$$\hat{z}_j = \sigma(\mathbf{x}) \cdot s_j + (1 - \sigma(\mathbf{x})) \cdot z_j, \tag{35}$$

where the confidence score function is σ,

$$s_j = \alpha_j \cdot z_j + \beta_j, \quad \forall j \in \mathscr{C}, \tag{36}$$

where α_j and β_j are learnable parameters of class j. The confidence score function $\sigma(\mathbf{x}) = g(\mathbf{v}^\top \mathbf{x})$ combines a linear layer and an activation function. In the second fine-tuning stage, the loss is designed bringing in a re-weighting factor

$$w_c = \frac{(1/n_c)^\rho}{\sum_{k=1}^{C}(1/r_k)^\rho}, \quad \forall c \in C, \tag{37}$$

where ρ is a hyper-parameter determining the encoding class prior. The total loss is

$$\begin{aligned} L &= \mathbb{E}_{D_{tr}}[KL(p_r(y|\mathbf{x}) \| p_m(y|\mathbf{x}))] \\ &\approx -\tfrac{1}{N}\sum_{i=1}^{N}\left[\sum_{y \in C} p_r(y|\mathbf{x}_i) \log\left(p_m(y|\mathbf{x}_i)\right)\right] + C, \end{aligned} \tag{38}$$

which aims at minimizing the KL divergence between reference distribution $p_r(y = c|\mathbf{x}_i) = w_c \cdot \delta_c(y_i), \forall c \in \mathscr{C}$ and logits distribution p_m. $\delta_c(y_i)$ is equal to 1 when $y_i = c$ while, in other cases, it equals to 0. DisAlign calibrates the logits distribution effectively by confidence-aware adapted distribution alignment.

When considering the over-fitting of head classes in long-tailed recognition, [73] derived a formula that could measure the complexity and biased decision boundary that each sample caused and proposed influence-balanced training (IB). The key insight is down-weighting the highly influential samples could smooth the decision boundary, and meanwhile, alleviate the over-fitting of majority classes. It combines two stages, which are normal training by random sampling method and CE loss, and fine-tuning for influence balancing. The loss in the second stage is shown as

$$L_{IB} = \sum_{(x,y) \in D} \lambda_k \frac{L(y, \delta(\mathbf{z}))}{\|\delta(\mathbf{z}) - y\|_1 \cdot \|\mathbf{x}\|_1}, \tag{39}$$

where $L(y, \delta(\mathbf{z})) = -\sum_{j=0}^{C} y_j \log \delta(z_j)$. This method attempts to alleviate over-fitting by down-weighting the influential samples.

In summary, methods in the fine-tuning stage follow a two-stage strategy that learns features with uniform sampling strategy and then fine-tunes classifier with different loss or sampling methods. The key insight of these methods comes from the empirical observation that random sampling from all training data yields the most generalizable feature representation compared with other sampling strategies. They usually yield competitive results. However, they also require a two-stage training procedure and cannot be easily combined with other techniques.

4.3. Inference stage

During inference, many strategies directly calibrate the logits distribution by introducing hyper-parameters to balance weights or removing bad effects in the biased learning process [40,66,95,96,126].

An observation is that the classifier weight correlates with the decision boundary, where the majority classes have larger weights and larger margins while the minorities are in the opposite situation. This observation provides a simple idea that controls the margins of the few-shot categories by weight normalization. For instance, WVN algorithm in [42] carry out in two steps. The first is training with cross-entropy loss, instance balanced sampling strategy and classifier weight vectors normalization by $\forall i, \boldsymbol{w}_i \leftarrow \frac{\boldsymbol{w}_i}{\|\boldsymbol{w}_i\|}$. The other is re-scaling the weights in the inference stage as $\forall i, \boldsymbol{w}_\mathbf{i} \leftarrow \left(\frac{n_1}{n_i}\right)^\gamma \boldsymbol{w}_i$. n_1 is the highest class frequency, n_i is the frequency of subset D_i and γ is a hyperparameter in the range $(0, 1)$.

Similarly, Kang et al. [40] have an empirical observation that the distribution of normalized logit weights in different categories is correlated with the long-tailed distribution after jointly training in instance-balanced sampling. Therefore, a straightforward idea is to regularize the logit weights by a hyperparameter τ. Such a τ-norm method balances the classifier weights $\boldsymbol{W}$ by $\widetilde{\boldsymbol{w}_i} = \boldsymbol{w}_i / \|\boldsymbol{w}_i\|^\tau$, where the τ controls the normalization temperature in each class i and $\|\cdot\|$ represents L_2 normalization. To rectify the weights smoothly, $\tau \in (0, 1)$ is used to avoid no-normalization at $\tau = 0$ and standard L_2 normalization at $\tau = 1$. The bias weight is ignored during normalization for the negligible influence on final results. The value of τ could be chosen by cross-validation.

Menon et al. [66] trained the backbone datasets in instance balanced sampling method by Adam optimizer, the logits weights are either anti-correlation or independent of the class frequencies, which presented the limitation of τ normalized classifier. To solve this problem, inconsistency of the margin loss such as hinge loss, the logit adjustment [66] was introduced. This adjusting method is based on Bayes-optimal prediction which gives the best estimation of label given the specific instance. Considering the multiclass classification problem, the goal of the long-tailed visual recognition is to minimize the balanced error

$$\mathrm{BER}(f) \doteq \frac{1}{C}\sum_{y\in Y} p_{\mathbf{z}|y}\left(y \notin \mathrm{argmax}_{y'\in \mathbb{y}} f_{y'}(\mathbf{z})\right). \tag{40}$$

$\mathbb{y}$ is the multiclass label. When $f^* \in \mathrm{argmin}_{f:\mathcal{X}\to\mathbb{R}^C}\mathrm{BER}(f)$, we get [10,67]

$$\begin{aligned}\mathrm{argmax}_{y\in Y} f_y^*(\mathbf{z}) &= \mathrm{argmax}_{y\in Y} p^{\mathrm{bal}}(y|\mathbf{z}) \\ &= \mathrm{argmax}_{y\in Y} p(\mathbf{z}|y).\end{aligned} \tag{41}$$

where the p^{bal} is the balanced class probability. Then, for fixed $p(\mathbf{z}|y)$, the change of $p(y)$ does not influence the optimal choice. According to the Bayes theorem, $p^{\mathrm{bal}}(y|\mathbf{z}) \propto p(y|\mathbf{z})/p(y)$, and suppose $p(y|\mathbf{z}) \propto \exp\left(g_y^*(\mathbf{z})\right)$. Then, Eq. 41 becomes

$$\begin{aligned}&\mathrm{argmax}_{y\in Y} p^{\mathrm{bal}}(y|\mathbf{z}) = \mathrm{argmax}_{y\in Y} \exp\left(g_y^*(\mathbf{z})\right)/p(y) \\ &= \mathrm{argmax}_{y\in Y} g_y^*(\mathbf{z}) - \ln p(y)\end{aligned} \tag{42}$$

Based on Eq. 42, the post hoc logit adjustment is

$$\begin{aligned}&\mathrm{argmax}_{y\in[C]} \exp\left(\boldsymbol{w}_y^\top \Phi(\mathbf{z})\right)/\pi_y^\tau \\ &= \mathrm{argmax}_{y\in[C]} f_y(\mathbf{z}) - \tau\cdot\log\pi_y,\end{aligned} \tag{43}$$

where the $\pi \in \Delta_y$ is the estimate of class prior probability. In the long-tailed problem, the class prior probabilities could be the class frequencies in training data. The hyperparameter $\tau > 0$. The performance in long-tailed datasets is slightly better than τ-norm classifier.

After investigating the related long-tailed paradox, a critical question is not settled: why these algorithms are practical to improve the overall or tail class performance. No fundamental theories are provided in these algorithms, including the simple but effective two-stage methods such as τ normalization and logits adjustment methods. Inspired by this, Tang et al. [95] attempted to tackle this issue from the perspective of causal inference. They investigated the bias in the training process from imbalanced training data, where the momentum effect M is influential and proposed the total direct effect inference (TDE) method.

The momentum optimization strategy implemented in Pytorch is described as:

$$v_t = \underbrace{\mu\cdot v_{t-1}}_{\text{momentum}} + g_t, \quad \theta_t = \theta_{t-1} - lr\cdot v_t, \tag{44}$$

where the notations in the t_{th} iteration are: model parameters θ_t, gradient g_t, velocity v_t, momentum decay ratio μ, and learning rate lr. The momentum accumulates the past gradients and determines the optimization direction. However, in long-tailed visual problems, the accumulated former directions increase the bias to head categories and worsen the test accuracy, especially the minorities. The TDE is described as

$$\begin{aligned}\arg\max_{i\in C}\mathrm{TDE}(Y_i) &= [Y_d = i|do(Z=\mathbf{z})] \\ &\quad -[Y_d = i|do(Z=\mathbf{z}_0)],\end{aligned} \tag{45}$$

where $\boldsymbol{z}_0$ is a null input (0 in this paper) and the *do*-operator is the causal intervention that results in $M\mapsto Z$ in the causal graph, which removes the bad effects. In the implementing stage, the method firstly calculates the head direction in the training process and removes the bad effect without fine-tuning. The final TDE method is

$$\mathrm{TDE}(Y_i) = \frac{\tau}{K}\sum_{k=1}^{K}\left(\frac{\left(\boldsymbol{w}_i^k\right)^\top \boldsymbol{z}^k}{\left(||\boldsymbol{w}_i^k||+\gamma\right)||\boldsymbol{z}^k||} - \alpha\cdot\frac{cos\left(\boldsymbol{z}^k,\hat{\boldsymbol{d}}^k\right)\cdot\left(\boldsymbol{w}_i^k\right)^\top\hat{\boldsymbol{d}}^k}{||\boldsymbol{w}_i^k||+\gamma}\right) \tag{46}$$

where $\hat{\boldsymbol{d}}^k$ is the head direction of group k, τ is a positive scaling factor and γ is a hyper-parameter controlling the normalization level.

In summary, the aforementioned inference stage methods share a similar two-stage training strategy adopted in fine-tuning stage methods. The difference is that the fine-tuning stage methods calibrate the bias via fine-tuning whereas the inference stage methods calibrate during inference. The inference stage methods are generally simple and easy to implement while also yielding competitive results. One possible drawback is the lack of flexibility at the inference stage.

5. Experiments and comparisons

In the preceding section, three types of methods for tackling long-tailed recognition issues are revisited. To get a clearer idea of how well the algorithms work, we compare the top-1 accuracy on three benchmarks: CIFAR-LT, ImageNet-LT, and iNaturalist (See 5.1). To evaluate the method's performance and adapt to various conditions, we generate long-tailed, step, linear test distributions (See 5.2). Finally, to get more comprehensive results, we introduce per-class accuracy, multi-label ROC AUC, and ECE evaluation measures as assessments in three types of distributions (Long-tailed, Linear and Step) (See 5.3). Two NVIDIA Tesla V100 GPU cards are used in this study.

5.1. Top-1 accuracy in balanced test set

We follow the prevalent setting where a balanced test distribution is used. In this section, methods are evaluated on three popular benchmarks CIFAR-LT, ImageNet-LT and iNaturalist (See 3.1) with top-1 accuracy (See 3.2). We choose ResNet-32, ResNeXt-50 and ResNet-50 (See 2.2) in experiments. The baseline models are trained with vanilla ResNet/ResNeXt models, random sampling method and SGD optimization strategy. The results are shown in Table 3, Table 5 and Table 6. For a fair comparison, most of the results are obtained from the corresponding papers. However, the training parameters such as learning rate and training epochs are not restricted because of different training strategies among methods. For example, the multiple experts and transfer learning methods generally require higher computational costs than other tricks. Results with † are re-implemented by our frameworks trained in batch size 128, SGD optimization method with initial learning rate 0.2, momentum 0.9 and weight decay 5×10^{-4} for 90 epochs. We set the scheduler epoch for 30 with $\gamma = 0.1$. The input images are of size 224×224 with random crop and flip.

The evaluation results illustrate that traditional methods introduced from the imbalanced problem, including re-sampling methods and basic cost-sensitive loss, have less prominent performance. However, in the category of multiple experts and transfer learning, cost-sensitive loss and fine-tuning stage, methods have relatively better results. Specifically, BALMS, PaCo and CBD-ens are state-of-the-art methods in CIFAR-10/100 im100, ImageNet-LT and iNaturalist benchmarks. MiSLAS and DRO-LT get the best results in CIFAR-10 and CIFAR-100 datasets with the imbalance rate of 50.

5.2. Comparison in uneven test distributions

Current methods are mainly evaluated on the balanced test set. However, in real scenarios, the prior distribution of categories is unknown. Naturally, test distribution may also exhibit long-tailed

Table 3
Top-1 accuracy evaluated in CIFAR-LT dataset with ResNet-32.

	CIFAR-LT 10		CIFAR-LT 100	
Methods	**im100**	**im50**	**im100**	**im50**
Baseline	69.8	75.2	38.3	42.1
CS loss [38]	70.9	76.3	29.1	36.2
CB Re-sampling [40]	69.6	76.0	32.7	38.5
SR Re-sampling [62,69]	68.6	75.2	35.5	40.2
Mixup [123]	73.1	77.8	39.6	45.0
Focal loss [51]	70.4	75.3	38.1	42.4
Manifold mixup [19]	73.0	78.0	38.3	43.1
PG Re-sampling [6,13]	67.1	75.0	38.6	42.9
CB-Focal [12]	74.6	79.3	39.6	45.2
CB loss [12]	74.7	79.3	39.6	45.3
Adaptive [6]	73.4	–	39.6	–
τ-norm($\tau = 1$) [40]	76.0	–	41.1	–
OLTR [60]	–	–	41.2	–
smDRAGON [84]	77.9	–	42.0	–
LDAM-DRW [6]	77.0	79.3	42.0	45.1
LA [66]	80.9	–	42.1	–
BBN [132]	79.8	82.2	42.6	47.0
ELF(LDAM)+DRW [19]	78.1	82.4	43.1	47.5
cRT [95]	82.0	–	43.3	–
EQL [92]	–	–	43.4	–
M2m [105]	79.1	–	43.5	–
LFME-LDAM [118]	–	–	43.8	–
CBA-LDAM [19]	80.0	82.2	44.1	49.2
De-confond TDE [95]	80.6	83.6	44.1	50.3
Hybrid-SC [105]	81.4	85.4	46.7	51.9
Remix-DRW [8]	79.8	–	46.8	–
MiSLAS [131]	82.1	**85.7**	47.0	52.3
LWS [40]	83.7	–	–	–
DRO-LT [85]	–	–	47.3	**57.6**
RIDE [107]	–	–	49.1	–
BALMS [78]	**84.9**	–	**50.8**	–

distribution following Zipf's law, while in extreme situations such as in wildlife reserves for animal recognition, the distribution might have a negative correlation with the long-tailed distribution. To further explore the adaption abilities of aforementioned methods, 14 distributions of long-tailed distribution, step distribution and linear distribution with various parameters are generated. Specifically, power value α and the reverse flag *Rev* are considered in a long-tailed setting. In step distribution, we take step number S, minimum class number C_{min} and the reverse flag *Rev* into consideration, while in linear setting, the minimum class number C_{min} and the reverse flag *Rev* are the parameters. The test images in different distributions are generated based on the OLTR [60] with 50 test instances each class. The test distributions of $Rev = 0$ are shown as Fig. 14 and we can get the reverse distributions by left–right flip.

To evaluate deep models performance, we select 15 methods from previously reviewed algorithms in three categories. The method selection from each category is based on the category proportion of the overall methods and on the performance. The comparing methods are baseline, OLTR, class-balanced re-sampling (CB RS), cRT, LWS, TDE, MiSLAS, RIDE(4 experts) and PaCo. We use the pre-trained model provided by the paper, while for TDE, OLTR and MiSLAS, we train models using the provided code and keep the parameter unchanged. The evaluation results and trends are shown in Table 4 and Fig. 10.

The results show that there is a relatively significant difference among varied test distributions. The fifth test set (Long-tailed with $\alpha = 2.0$ and $Rev = False$) has the overall highest top-1 accuracy, while the ninth test set is the lowest. Take PaCo as an example, the difference between these two test sets is 48.03%. The accuracy in reverse distributions is considerably lower than the opposite.

Table 4
Top-1 accuracy of ImageNet-LT dataset evaluated in different test distributions.

Test Set	ID	Params	Baseline	OLTR [60]	CB RS [40]	PG RS [6,13]	cRT [40]	LWS [40]	TDE [95]	MiSLAS [131]	RIDE [107]	PaCo [11]
Even	**1**	–	44.36	38.37	45.07	47.16	49.62	49.92	50.48	53.49	56.88	**58.31**
LT	**2**	$\alpha = 0.1$ $Rev = 0$	54.11	42.25	52.29	53.85	55.38	54.86	55.35	59.23	61.85	**62.26**
	3	$\alpha = 0.5$ $Rev = 0$	57.59	42.51	54.43	55.63	57.05	56.12	56.63	60.62	63.74	**63.98**
	4	$\alpha = 1.0$ $Rev = 0$	61.00	44.89	56.90	58.50	59.48	58.61	58.94	63.66	**66.20**	66.14
	5	$\alpha = 2.0$ $Rev = 0$	65.99	47.36	60.38	61.87	62.55	60.50	61.35	66.16	**68.67**	67.43
	6	$\alpha = 0.1$ $Rev = 1$	33.30	33.04	36.45	39.22	43.14	44.60	45.14	47.00	50.72	**52.82**
	7	$\alpha = 0.5$ $Rev = 1$	29.14	31.12	33.15	35.97	40.62	41.76	42.49	44.53	48.28	**49.77**
	8	$\alpha = 1.0$ $Rev = 1$	25.64	28.74	29.85	33.13	37.80	39.41	40.46	41.79	45.73	**47.22**
	9	$\alpha = 2.0$ $Rev = 1$	17.97	23.22	23.17	26.40	32.27	34.22	35.68	36.37	40.47	**42.18**
Linear	**10**	$Min = 5$ $Rev = 0$	54.52	42.68	52.96	54.20	55.42	54.93	55.59	59.19	62.42	**63.12**
	11	$Min = 5$ $Rev = 1$	34.36	33.86	37.02	39.92	44.05	44.97	45.49	47.58	51.53	**53.64**
Step	**12**	$S = 2$ $Rev = 0$ $Min = 10$	56.95	43.40	54.81	55.93	56.75	55.75	56.57	60.57	63.17	**64.05**
	13	$S = 2$ $Rev = 1$ $Min = 10$	32.40	33.83	35.54	38.67	42.86	44.22	44.69	46.39	50.84	**52.62**
	14	$S = 3$ $Rev = 0$ $Min = 10$	55.48	42.98	53.60	54.94	55.83	55.29	56.11	60.26	62.71	**63.68**
	15	$S = 3$ $Min = 10$ $Rev = 1$	33.43	33.78	36.43	39.58	43.46	44.52	44.92	46.91	50.75	**53.20**

Table 5
Top-1 accuracy evaluated in ImageNet-LT dataset with ResNeXt-50.

Methods	Many	Median	Low	Top-1
Baseline	65.9	37.5	7.7	44.4
Focal Loss [51]	64.3	37.1	8.2	43.7
CB Re-sampling [40]	61.8	40.1	15.5	45.1
SR Re-sampling [62,69]	64.3	41.2	17.0	46.8
PG Re-sampling [6,13]	61.9	43.2	19.4	47.2
NCM [40]	56.6	45.3	28.1	47.3
τ-norm [40]	59.1	46.9	30.7	49.4
cRT [40]	61.8	46.2	27.4	49.6
LWS [40]	60.2	47.2	30.3	49.9
Seasaw Loss [104]	67.1	45.2	21.4	50.4
De-confound-TDE [95]	62.7	48.8	31.6	51.8
DisAlign [33]	62.7	52.1	31.4	53.4
PaCo [11]	**67.5**	**56.9**	**36.7**	**58.2**

Table 6
Top-1 accuracy evaluated in iNaturalist dataset with ResNet-50.

Methods	Top-1	Methods	Top-1
Baseline†	60.8	LWS [40]	69.5
CB-Focal [105]	61.1	BBN [132]	69.6
CB Re-sampling†[40]	62.0	DRO-LT [85]	69.7
NCM [40]	63.1	Hybrid-PSC [105]	70.4
OLTR [60]	63.9	Remix [8]	70.5
FSA [9]	65.9	DisAlign [33]	70.6
LA [66]	66.3	GistNet [54]	70.8
cRT [40]	67.6	MiSLAS [131]	71.6
LDAM [6]	68.0	RIDE [33]	72.6
smDRAGON [84]	69.1	PaCo [11]	73.2
τ-norm [40]	69.3	CBD-ens [36]	**73.6**

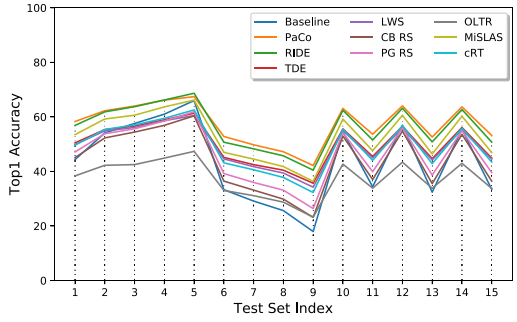

Fig. 10. Top-1 accuracy of ImageNet-LT evaluated in varied test distributions. X-axis is the test set index in Table 4 with different test set distributions and y-axis is the top-1 accuracy.

The accuracy of the long-tailed test distributions without reverse is higher than the balanced test set. This performance difference stems from the accuracy bias among head and tail classes, which is proved by the performance gap between many-shot and low-shot classes in Table 5.

5.3. Evaluation metrics robustness

Previous methods are commonly evaluated using top-1 accuracy for overall performance. However, it is sensitive to test distributions. For example, facing the long-tailed distribution of test set without reverse, the dominant head classes will render biased test accuracy. Therefore, we introduce per-class accuracy (P-C A) as an additional evaluation method to treat each class equally and avoid bias. Two popular evaluation metrics, multi-class ROC AUC one-vs-rest (AUC) and Expected Calibration Error (ECE) in percentage are also adopted. The former evaluate algorithms by TPR and FPR over the threshold range $[0, 1]$. The evaluation results are shown in Table 7 and illustrated in Figs. 11–13.

The results show that PaCo performs the best in terms of per-class accuracy and multi-class ROC AUC evaluation metrics. For ECE (lower is better), RIDE has significantly lower values than other algorithms, while OLTR gets the highest error. Besides, similar to top-1 accuracy, the reversed test distributions are lower than the counterpart. When evaluated in Multi-class ROC AUC, RIDE has relatively poor performance than other methods.

6. Challenges, outlook and conclusion

6.1. Challenges

While recent years have witnessed a significant progress on long-tailed visual recognition, the research of this area is still in its infancy and is largely challenged by multiple issues. The challenges include the overall performance, model and training complexity, and the robustness to unknown test distributions, each being elaborated below.

Firstly, the model performance still has a large room for improvement. Current state-of-the-art methods achieved 58.3% and 73.6% accuracy in ImageNet-LT and iNaturalist benchmarks, and surpass the plain model by a large margin (over 10%). However, if we compare them with their counterparts on full ImageNet, e.g. ResNet-50 yields 78.57% in Top-1 accuracy, there is still a huge gap of over 20%. The performance on the full ImageNet can be regarded as an upper bound performance, and an ideal long-tailed recognition algorithm would approach the upper bound performance under the less and long-tailed data. Therefore, a lot of efforts are still needed to further improve the model performance.

Secondly, the improvement in prediction accuracy usually comes at a cost of increasing model complexity. This would raise the difficulty of deployment in real-world applications, especially on embedded and mobile devices. Simple strategies, such as re-sampling and cost-sensitive learning methods, have limited performance gain, whereas complex methods, including transfer learning and multi-stage methods, usually end up with superior performance with an enormous computational cost. Hence, how to trade off between a lightweight model and high performance, especially in the context of practical applications, becomes one challenging issue in the field.

Finally, the model robustness against different test set distributions has been under-explored. In real-world applications, test set distribution is very much dependent on the specific task and usually varies significantly. However, most long-tailed recognition methods are evaluated under the balanced test set assumption. Whether the recent progress on the balanced test set also applies to other testing distributions, remains an open question. In other words, it is not very clear whether existing methods are really solving the long-tailed problem, or they just simply over-compensate for the tail class to cater for the balanced test set.

6.2. Outlook

Having analyzed and evaluated the aforementioned methods, we further discuss possible and potential future directions.

- **More factors in data construction**: Data is the foundation of deep neural networks training. However, constructing a high-quality dataset would require a large amount of human-labeled data, which is expensive. One natural question would be, how to evaluate the cost of collecting images and annotating a dataset, especially in the context of long-tailed distribution. For instance, collecting and annotating a head category, such

Table 7
Evaluation results of ImageNet-LT dataset with three evaluation methods.

Test set	Params	Eval	Baseline	OLTR [60]	CB RS [40]	PG RS [6,13]	cRT [40]	LWS [40]	TDE [95]	MiSLAS [131]	RIDE [107]	PaCo [11]
Even	–	P-C A	44.36	38.37	45.07	47.16	49.62	49.92	50.48	53.49	56.88	**58.31**
		ECE	10.20	17.73	10.49	10.75	10.93	12.36	11.71	12.05	**3.62**	11.29
		AUC	98.80	97.89	98.42	98.66	98.84	98.97	98.82	99.02	97.07	**99.28**
LT	$\alpha = 0.1$	P-C A	20.66	16.13	19.96	20.56	21.14	20.94	21.13	22.61	23.61	**23.77**
	Rev = 0	ECE	10.72	19.39	11.82	11.68	11.36	12.67	11.95	12.76	**3.51**	10.71
		AUC	99.02	98.03	98.61	98.86	99.00	99.09	98.96	99.11	97.27	**99.37**
	$\alpha = 0.5$	P-C A	16.50	12.18	15.59	15.94	16.34	16.08	16.22	17.37	18.26	**18.33**
	Rev = 0	ECE	10.74	18.98	12.12	11.41	11.07	12.23	11.62	12.80	**3.54**	10.76
		AUC	98.94	97.91	98.44	98.77	98.90	99.05	98.89	99.04	97.44	**99.32**
	$\alpha = 1.0$	P-C A	12.86	9.46	12.00	12.33	12.54	12.36	12.43	13.42	**13.96**	13.94
	Rev = 0	ECE	10.49	20.32	12.07	11.69	10.78	12.28	11.62	12.94	**3.11**	10.58
		AUC	99.02	97.83	98.58	98.81	98.97	99.06	98.93	99.03	97.49	**99.37**
	$\alpha = 2.0$	P-C A	8.58	6.16	7.85	8.04	8.13	7.86	7.97	8.60	**8.93**	8.76
	Rev = 0	ECE	10.14	20.78	12.18	11.46	10.80	12.03	12.17	13.17	**3.14**	9.95
		AUC	99.05	97.89	98.53	98.76	98.95	98.92	98.80	98.90	97.40	**99.29**
	$\alpha = 0.1$	P-C A	12.71	12.61	13.91	14.97	16.47	17.03	17.23	17.94	19.36	**20.16**
	Rev = 1	ECE	8.82	15.42	9.25	9.69	10.60	12.19	11.61	11.44	**3.90**	11.48
		AUC	98.74	97.89	98.31	98.62	98.81	98.95	98.80	99.01	96.89	**99.25**
	$\alpha = 0.5$	P-C A	8.35	8.91	9.50	10.31	11.64	11.96	12.17	12.76	13.83	**14.26**
	Rev = 1	ECE	8.02	14.84	8.34	9.13	10.33	11.72	11.43	11.49	**3.99**	11.31
		AUC	98.63	97.80	98.23	98.52	98.72	98.89	98.71	98.93	96.54	**99.23**
	$\alpha = 1.0$	P-C A	5.41	6.06	6.29	6.98	7.97	8.31	8.53	8.81	9.64	**9.95**
	Rev = 1	ECE	7.69	13.35	7.95	8.77	9.92	11.23	11.41	11.06	**3.78**	11.20
		AUC	98.65	97.80	98.24	98.53	98.77	98.85	98.74	98.91	96.53	**99.18**
	$\alpha = 2.0$	P-C A	2.34	3.02	3.01	3.43	4.19	4.45	4.64	4.73	5.26	**5.48**
	Rev = 1	ECE	6.15	10.81	6.66	7.42	9.66	10.39	10.51	10.21	**3.76**	11.42
		AUC	98.35	97.71	97.99	98.26	98.51	98.72	98.65	98.84	96.26	**99.10**
Linear	$Min = 5$	P-C A	30.53	23.90	29.65	30.35	31.03	30.76	31.13	33.14	34.95	35.34
	Rev = 0	ECE	11.21	19.29	11.78	11.57	11.16	12.58	11.85	12.39	**3.48**	10.91
		AUC	98.96	98.01	98.55	98.77	98.92	99.03	98.91	99.07	97.31	**99.33**
	$Min = 5$	P-C A	19.24	18.96	20.73	22.35	24.67	25.18	25.47	26.64	28.85	**30.03**
	Rev = 1	ECE	9.30	15.90	9.11	9.78	10.90	12.11	11.77	11.61	**4.02**	11.67
		AUC	98.70	97.87	98.30	98.59	98.79	98.93	98.74	98.99	96.82	**99.25**
Step	$Min = 10$ $S = 2$	P-C A	34.17	26.04	32.89	33.56	34.05	33.45	33.94	36.34	37.90	**38.43**
	Rev = 0	ECE	11.50	19.72	12.12	11.86	11.41	12.66	11.55	12.62	**3.30**	11.01
		AUC	98.92	97.91	98.54	98.73	98.90	99.00	98.88	99.06	97.41	**99.31**
	$Min = 10$ $S = 2$	P-C A	19.44	20.30	21.32	23.20	25.72	26.53	26.82	27.83	30.50	**31.57**
	Rev = 1	ECE	9.49	16.08	9.02	9.80	11.00	12.26	11.79	11.44	**3.84**	11.55
		AUC	98.73	97.92	98.31	98.59	98.81	98.95	98.79	99.02	96.85	**99.23**
	$Min = 10$ $S = 3$	P-C A	33.29	25.79	32.16	32.97	33.50	33.17	33.67	36.16	37.62	**38.21**
	Rev = 0	ECE	11.17	19.54	11.62	11.60	11.04	12.55	11.81	12.76	**3.41**	10.90
		AUC	98.92	97.90	98.48	98.75	98.91	99.00	98.88	99.06	97.24	**99.30**
	$Min = 10$ $S = 3$	P-C A	20.06	20.27	21.86	23.75	26.08	26.71	26.95	28.14	30.45	**31.92**
	Rev = 1	ECE	9.29	15.84	9.12	9.96	10.82	12.14	11.68	11.46	**3.84**	11.68
		AUC	98.67	97.80	98.29	98.52	98.77	98.91	98.76	98.97	96.82	**99.22**

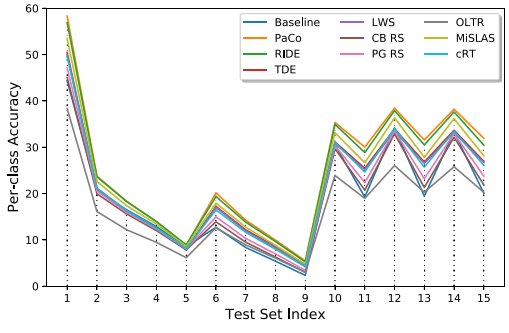

Fig. 11. Per-class accuracy of ImageNet-LT evaluated in varied test distributions. X-axis is the test set index in Table 4 with different test set distributions and y-axis is the per.-class accuracy.

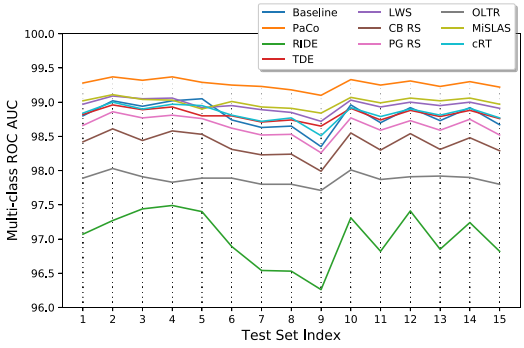

Fig. 12. Multi-class ROC AUC results of ImageNet-LT evaluated in varied test distributions. X-axis is the test set index in Table 4 with different test set distributions and y-axis is the Multi-class ROC AUC value in percentage.

as cat, is easy, while collecting and annotating a tail class, such as armadillo, would be more time-consuming. However, collecting cat images would bring only little benefit for the recognition performance, while collecting more armadillo images could largely improve the recognition accuracy on armadillo. Thus, how to make a trade-off between performance gain and dataset construction cost is crucial. In other words, how many training samples are needed for each class before the performance gain is saturated could be investigated. We believe that taking the per-class data construction cost as well as the expected performance gain into consideration will guide us to build a dataset more efficiently.

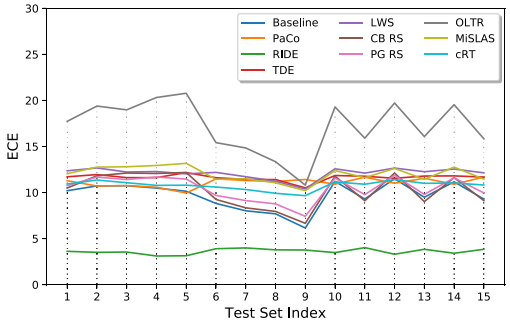

Fig. 13. ECE result of ImageNet-LT evaluated in varied test distributions. X-axis is the test set index in Table 4 with different test set distributions and y-axis is the ECE value.

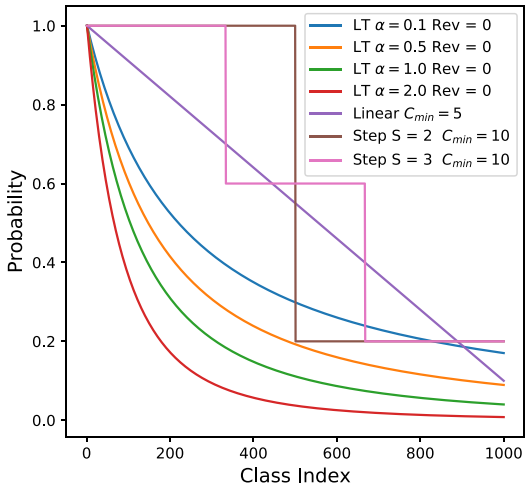

Fig. 14. Test distributions with $Rev = 0$. The x-axis is the class index in probability descending order and the y-axis is the probability of the selected instance number against the original test image number.

- **More factors in long-tailed training**: Apparently, a crucial step of long-tailed training is to recognize where head classes are and where tail classes are. Current long-tailed recognition methods only count *number of instances per class* to determine head/easy and tail/hard classes. However, there are certainly more factors that could potentially indicate the difficulty of a long-tailed recognition task. For instance, the intra-class diversity. A class with diverse training samples is easier to be classified and is expected to have higher accuracy. Moreover, the difficulty of the category itself also affects the performance. Therefore, taking more factors into consideration may help better determine head and tail classes.
  Apart from the factors of each category itself, the inter-class relationships could also be exploited. For instance, a minority class *Tortoise* usually appears by the *Stone* or *grass*, both of which are majority classes. Therefore, recognizing *Stone* and *grass* could be helpful to identify *Tortoise*. That is, exploiting such co-existing relationships as well as their semantic meanings could aid in recognizing those classes with few examples. In future, knowledge graphs and relevant techniques might be a potential solution.
- **More factors in long-tailed evaluation**: As mentioned before, most existing methods assume the testing set is balanced, and use Top-1 accuracy as the evaluation metric. On the one hand, the performance under arbitrary testing distribution could be investigated. On the other hand, more metrics, rather than Top-1 accuracy, could be considered. One may consider incorporating the per-class misclassification cost during evaluation. In many applications, the costs of misclassifying different classes are different, and the cost-aware evaluation protocol might enable a fair evaluation.
- **More training settings with long-tailed recognition.** Most existing long-tailed recognition methods are conducted in a fully supervised setting. Although few attempts have been made on semi-supervised or self-supervised learning for long-tailed recognition, more settings and realistic scenarios could be investigated. For instance, one could borrow a related auxiliary dataset and boost the long-tailed recognition performance. One could also exploit the search engine and collect webly-labeled data at low cost, and use those webly-supervised, or weakly supervised data to improve the long-tailed recognition. Moreover, since the long-tailed visual recognition problem stems from the long-tailed distribution of natural concepts, incorporating multiple modalities, such as text with semantic concepts, could be helpful. Finally, active learning could be incorporated so that tail class instances can be queried with human-in-the-loop.

6.3. Conclusion

In this review, methods for solving the long-tailed visual recognition problem have been discussed. These approaches are divided into three categories, highlighting their respective contributions. We described key algorithmic contributions of the typical techniques and summarized their strengths and weaknesses. To better understand different strategies, we validated the contributions of more than ten strategies on various test distributions using diverse assessment methods. Finally, we suggested several intellectual challenges and potential ideas that need to be studied in the future.

CRediT authorship contribution statement

Yu Fu: Methodology, Software, Validation, Writing - original draft. **Liuyu Xiang:** Conceptualization, Methodology, Writing - original draft. **Yumna Zahid:** Methodology, Investigation, Writing - review & editing. **Guiguang Ding:** Conceptualization, Writing - review & editing, Supervision. **Tao Mei:** Conceptualization, Writing - review & editing, Supervision. **Qiang Shen:** Conceptualization, Writing - review & editing, Supervision. **Jungong Han:** Conceptualization, Writing - review & editing, Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Y. Bengio, A.C. Courville, P. Vincent, Representation learning – a review and new perspectives, in: IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2013.
[2] P. Branco, L. Torgo, R.P. Ribeiro, A survey of predictive modeling on imbalanced domains, ACM Computing Surveys (CSUR) 49 (2016) 1–50.
[3] M. Buckland, F. Gey, The relationship between recall and precision, J. Am. Soc. Inform. Sci. 45 (1994) 12–19.
[4] M. Buda, A. Maki, M.A. Mazurowski, A systematic study of the class imbalance problem in convolutional neural networks, Neural Networks (2018).
[5] D. Cao, X. Zhu, X. Huang, J. Guo, Z. Lei, Domain balancing - face recognition on long-tailed domains, Computer Vision and Pattern Recognition (CVPR), 2020.
[6] K. Cao, C. Wei, A. Gaidon, N. Aréchiga, T. Ma, Learning imbalanced datasets with label-distribution-aware margin loss, in: Conference on Neural Information Processing Systems (NeurIPS), 2019.
[7] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, Smote: synthetic minority over-sampling technique, J. Artif. Intell. Res. 16 (2002) 321–357.
[8] Chou, H.P., Chang, S.C., Pan, J.Y., Wei, W., Juan, D.C., 2020. Remix - rebalanced mixup, in: European Conference on Computer Vision (ECCV).

[9] Chu, P., Bian, X., Liu, S., Ling, H., 2020. Feature space augmentation for long-tailed data, in: European Conference on Computer Vision (ECCV).
[10] Collell, G., Prelec, D., Patil, K., Reviving threshold-moving: a simple plug-in bagging ensemble for binary and multiclass imbalanced data arXiv:1606.08698.
[11] J. Cui, Z. Zhong, S. Liu, B. Yu, J. Jia, Parametric contrastive learning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 715–724.
[12] Y. Cui, M. Jia, T.Y. Lin, Y. Song, S. Belongie, Class-balanced loss based on effective number of samples, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 9268–9277.
[13] Y. Cui, Y. Song, C. Sun, A. Howard, S.J. Belongie, Large scale fine-grained categorization and domain-specific transfer learning, Computer Vision and Pattern Recognition (CVPR), 2018.
[14] Y. Cui, F. Zhou, J. Wang, X. Liu, Y. Lin, S. Belongie, Kernel pooling for convolutional neural networks, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 3049–3058, https://doi.org/10.1109/CVPR.2017.325.
[15] J. Deng, W. Dong, R. Socher, L.J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE conference on computer vision and pattern recognition, Ieee, 2009, pp. 248–255.
[16] J. Deng, J. Guo, N. Xue, S. Zafeiriou, Arcface: Additive angular margin loss for deep face recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 4690–4699.
[17] Z. Deng, H. Liu, Y. Wang, C. Wang, Z. Yu, X. Sun, Pml: Progressive margin loss for long-tailed age classification, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 10503–10512.
[18] A. Dubey, O. Gupta, R. Raskar, N. Naik, Maximum entropy fine-grained classification, in: Proceedings of the 32nd International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, 2018, pp. 635–645.
[19] Duggal, R., Freitas, S., Dhamnani, S., Horng, D., Sun, J., ELF: An early-exiting framework for long-tailed classification.
[20] S. Escalera, J. Fabian, P. Pardo, X. Baró, J. Gonzalez, H.J. Escalante, D. Misevic, U. Steiner, I. Guyon, Chalearn looking at people 2015: Apparent age and cultural event recognition datasets and results, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2015, pp. 1–9.
[21] J. Fu, H. Zheng, T. Mei, Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 4476–4484, https://doi.org/10.1109/CVPR.2017.476.
[22] S. Gidaris, P. Singh, N. Komodakis, Unsupervised representation learning by predicting image rotations, in: International Conference on Learning Representations (ICLR), 2018.
[23] K. Greff, A. Rasmus, M. Berglund, T. Hao, H. Valpola, J. Schmidhuber, Tagger: Deep unsupervised perceptual grouping, Adv. Neural Inform. Process. Syst. (2016) 4484–4492.
[24] Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q., 2017a. On calibration of modern neural networks, in: Proceedings of the 34th International Conference on Machine Learning – Volume 70, JMLR.org. p. 1321–1330.
[25] H. Guo, Y. Li, J. Shang, M. Gu, H. Yuanyue, G. Bing, Learning from class-imbalanced data - review of methods and applications, Expert Syst. Appl. (2017).
[26] H. Guo, S. Wang, Long-tailed multi-label visual recognition by collaborative training on uniform and re-balanced samplings, Computer Vision and Pattern Recognition (CVPR), 2021.
[27] Y. Guo, L. Zhang, Y. Hu, X. He, J. Gao, Ms-celeb-1m: A dataset and benchmark for large-scale face recognition, European conference on computer vision, Springer (2016) 87–102.
[28] A. Gupta, P. Dollér, R.B. Girshick, Lvis - a dataset for large vocabulary instance segmentation, Computer Vision and Pattern Recognition (CVPR), 2019.
[29] H. He, E.A. Garcia, Learning from imbalanced data, IEEE Trans. Knowl. Data Eng. 21 (2009) 1263–1284.
[30] H. He, Y. Ma, Imbalanced learning: foundations, algorithms, and applications, 2013.
[31] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
[32] X. Hong, S. Chen, C.J. Harris, A kernel-based two-class classifier for imbalanced data sets, IEEE Trans. Neural Networks 18 (2007) 28–41.
[33] Y. Hong, S. Han, K. Choi, S. Seo, B. Kim, B. Chang, Disentangling label distribution for long-tailed visual recognition, Computer Vision and Pattern Recognition (CVPR), 2021.
[34] C. Huang, Y. Li, C.C. Loy, X. Tang, Learning deep representation for imbalanced classification, Computer Vision and Pattern Recognition (CVPR), 2016.
[35] iNatrualist, The iNaturalist 2018 competition dataset.
[36] Iscen, A., Araujo, A., Gong, B., Schmid, C., Class-balanced distillation for long-tailed visual recognition arXiv:2104.05279.
[37] M.A. Jamal, M. Brown, M.H. Yang, L. Wang, B. Gong, Rethinking class-balanced methods for long-tailed visual recognition from a domain adaptation perspective, Computer Vision and Pattern Recognition (CVPR), 2020.
[38] N. Japkowicz, S. Stephen, The class imbalance problem - a systematic study, Intell. Data Anal. (2002).
[39] Kang, B., Li, Y., Yuan, Z., Feng, J., Exploring balanced feature spaces for representation learning, 15.
[40] B. Kang, S. Xie, M. Rohrbach, Z. Yan, A. Gordo, J. Feng, Y. Kalantidis, Decoupling representation and classifier for long-tailed recognition, in: International Conference on Learning Representations (ICLR), 2020.
[41] S.H. Khan, M. Hayat, S.W. Zamir, J. Shen, L. Shao, Striking the right balance with uncertainty, Computer Vision and Pattern Recognition (CVPR), 2019.
[42] Kim, B., Kim, J., Adjusting decision boundary for class imbalanced learning 8, 81674–81685. doi: 10.1109/ACCESS.2020.2991231. conference Name: IEEE Access.
[43] Kim, J., Jeong, J., Shin, J., M2m: Imbalanced classification via major-to-minor translation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13896–13905.
[44] S. Kong, C. Fowlkes, Low-rank bilinear pooling for fine-grained classification, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 365–374.
[45] B. Krawczyk, Learning from imbalanced data – open challenges and future directions, Progr. Artif. (2016), Intelligence.
[46] Krizhevsky, A., Hinton, G., et al., 2009. Learning multiple layers of features from tiny images.
[47] B.M. Lake, R. Salakhutdinov, J.B. Tenenbaum, Human-level concept learning through probabilistic program induction, Science 350 (2015) 1332–1338.
[48] C.H. Lampert, H. Nickisch, S. Harmeling, Learning to detect unseen object classes by between-class attribute transfer, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2009, pp. 951–958.
[49] S. Li, K. Gong, C.H. Liu, Y. Wang, F. Qiao, X. Cheng, Metasaug – meta semantic augmentation for long-tailed visual recognition, Computer Vision and Pattern Recognition (CVPR), 2021.
[50] T. Li, L. Wang, G. Wu, Self supervision to distillation for long-tailed visual recognition, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 630–639.
[51] T.Y. Lin, P. Goyal, R.B. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2020.
[52] T.Y. Lin, A. RoyChowdhury, S. Maji, Bilinear cnn models for fine-grained visual recognition, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 1449–1457.
[53] Ling, C.X., Li, C., 1998. Data mining for direct marketing: Problems and solutions, in: Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, AAAI Press. p. 73–79.
[54] Liu, B., Li, H., Kang, H., Hua, G., Vasconcelos, N., a. GistNet: a geometric structure transfer network for long-tailed recognition arXiv:2105.00131.
[55] Liu, B., Li, H., Kang, H., Vasconcelos, N., Hua, G., b. Semi-supervised long-tailed recognition using alternate sampling arXiv:2105.00133.
[56] J. Liu, Y. Sun, C. Han, Z. Dou, W. Li, Deep representation learning on long-tailed data – a learnable embedding augmentation perspective, Computer Vision and Pattern Recognition (CVPR), 2020.
[57] Liu, L., Liu, L., Investigate the essence of long-tailed recognition from a unified perspective arXiv:2107.03758.
[58] X. Liu, S. Li, M. Kan, J. Zhang, S. Wu, W. Liu, H. Han, S. Shan, X. Chen, Agenet: Deeply learned regressor and classifier for robust apparent age estimation, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2015, pp. 16–24.
[59] Z. Liu, P. Luo, X. Wang, X. Tang, Deep learning face attributes in the wild, in: 2015 IEEE International Conference on Computer Vision, ICCV 2015, IEEE Computer Society, 2015, pp. 3730–3738, https://doi.org/10.1109/ICCV.2015.425.
[60] Z. Liu, Z. Miao, X. Zhan, J. Wang, B. Gong, S.X. Yu, Large-scale long-tailed recognition in an open world, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2537–2546.
[61] M. Maalouf, T.B. Trafalis, Robust weighted kernel logistic regression in imbalanced and rare events data, Comput. Stat. Data Anal. 55 (2011) 168–183.
[62] Mahajan, D., Girshick, R.B., Ramanathan, V., He, K., Paluri, M., Li, Y., Bharambe, A., van der Maaten, L., 2018. Exploring the limits of weakly supervised pretraining, in: European Conference on Computer Vision (ECCV).
[63] Maloof, M.A., 2003. Learning when data sets are imbalanced and when costs are unequal and unknown, in: ICML-2003 workshop on learning from imbalanced data sets II, pp. 2–1.
[64] M.A. Mazurowski, P.A. Habas, J.M. Zurada, J.Y. Lo, J.A. Baker, G.D. Tourassi, Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance, Neural Networks 21 (2008) 427–436.
[65] D. Mease, A.J. Wyner, A. Buja, Boosted classification trees and class probability/quantile estimation, J. Mach. Learn. Res. 8 (2007).
[66] A.K. Menon, S. Jayasumana, A.S. Rawat, H. Jain, A. Veit, S. Kumar, Long-tail learning via logit adjustment, in: International Conference on Learning Representations (ICLR), 2021.
[67] Menon, A.K., Narasimhan, H., Agarwal, S., Chawla, S., On the statistical consistency of algorithms for binary classification under class imbalance, 9.
[68] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Conference on Neural Information Processing Systems (NeurIPS), 2013.
[69] More, A., Survey of resampling techniques for improving classification performance in unbalanced datasets arXiv:1608.06048.
[70] M.P. Naeini, G. Cooper, M. Hauskrecht, Obtaining well calibrated probabilities using bayesian binning, in: Twenty-Ninth AAAI Conference on Artificial Intelligence, 2015.

[71] S.S. Nath, G. Mishra, J. Kar, S. Chakraborty, N. Dey, A survey of image classification methods and techniques, in: 2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2014.
[72] G. Panis, A. Lanitis, N. Tsapatsoulis, T.F. Cootes, Overview of research on facial ageing using the fg-net ageing database, Iet Biometrics 5 (2016) 37–46.
[73] S. Park, J. Lim, Y. Jeon, J.Y. Choi, Influence-balanced loss for imbalanced visual classification, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2021, pp. 735–744.
[74] Peng, M., Zhang, Q., Xing, X., Gui, T., Huang, X., Jiang, Y.G., Ding, K., Chen, Z., Trainable undersampling for class-imbalance learning 33, 4707–4714. doi: 10.1609/aaai.v33i01.33014707.
[75] Ravi, S., Larochelle, H., 2016. Optimization as a model for few-shot learning.
[76] W.J. Reed, The pareto, zipf and other power laws, Econ. Lett. 74 (2001) 15–19.
[77] Ren, J., Yu, C., Cai, Z., Zhao, H., 2020a. Balanced activation for long-tailed visual recognition. arXiv.
[78] J. Ren, C. Yu, S. Sheng, X. Ma, H. Zhao, S. Yi, H. Li, Balanced meta-softmax for long-tailed visual recognition, in: Conference on Neural Information Processing Systems (NeurIPS), 2020.
[79] Ren, M., Zeng, W., Yang, B., Urtasun, R., 2018. Learning to reweight examples for robust deep learning, in: International Conference on Machine Learning (ICML).
[80] Ricanek, K., Tesafaye, T., 2006. Morph: A longitudinal image database of normal adult age-progression, in: 7th International Conference on Automatic Face and Gesture Recognition (FGR06), IEEE. pp. 341–345.
[81] D. Rodriguez, I. Herraiz, R. Harrison, J. Dolado, J.C. Riquelme, Preliminary comparison of techniques for dealing with imbalance in software defect prediction, in: Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering, 2014, pp. 1–10.
[82] R. Rothe, R. Timofte, L. Van Gool, Deep expectation of real and apparent age from a single image without facial landmarks, Int. J. Comput. Vision 126 (2018) 144–157.
[83] A. Sahoo, A. Singh, R. Panda, R. Feris, A. Das, Mitigating dataset imbalance via joint generation and classification, European Conference on Computer Vision, Springer (2020) 177–193.
[84] D. Samuel, Y. Atzmon, G. Chechik, From generalized zero-shot learning to long-tail with class descriptors, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2021, pp. 286–295.
[85] D. Samuel, G. Chechik, Distributional robustness loss for long-tail learning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2021.
[86] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, T. Lillicrap, Meta-learning with memory-augmented neural networks, International conference on machine learning, PMLR. (2016) 1842–1850.
[87] J. Schmidhuber, A neural network that embeds its own meta-levels, IEEE International Conference on Neural Networks, IEEE. (1993) 407–412.
[88] J. Shu, Q. Xie, L. Yi, Q. Zhao, S. Zhou, Z. Xu, D. Meng, Meta-weight-net – learning an explicit mapping for sample weighting, in: Conference on Neural Information Processing Systems (NeurIPS), 2019.
[89] Y. Sun, M.S. Kamel, A.K. Wong, Y. Wang, Cost-sensitive boosting for classification of imbalanced data, Pattern Recogn. 40 (2007) 3358–3378.
[90] Y. Sun, A.K.C. Wong, M.S. Kamel, Classification of imbalanced data - a review, in: International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI), 2009.
[91] F. Sung, Y. Yang, L. Zhang, T. Xiang, P.H. Torr, T.M. Hospedales, Learning to compare: Relation network for few-shot learning, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 1199–1208.
[92] Tan, J., Lu, X., Zhang, G., Yin, C., Li, Q., Equalization loss v2: A new gradient balance approach for long-tailed object detection arXiv:2012.08548.
[93] J. Tan, C. Wang, B. Li, Q. Li, W. Ouyang, C. Yin, J. Yan, Equalization loss for long-tailed object recognition, Computer Vision and Pattern Recognition (CVPR), 2020.
[94] Z. Tan, J. Wan, Z. Lei, R. Zhi, G. Guo, S.Z. Li, Efficient group-n encoding and decoding for facial age estimation, IEEE Trans. Pattern Anal. Mach. Intell. 40 (2017) 2610–2623.
[95] K. Tang, J. Huang, H. Zhang, Long-tailed classification by keeping the good and removing the bad momentum causal effect, in: Conference on Neural Information Processing Systems (NeurIPS), 2020.
[96] J. Tian, Y.C. Liu, N. Glaser, Y.C. Hsu, Z. Kira, Posterior re-calibration for imbalanced datasets, in: Conference on Neural Information Processing Systems (NeurIPS), 2020.
[97] K.M. Ting, A comparative study of cost-sensitive boosting algorithms, in: Proceedings of the 17th International Conference on Machine Learning, Citeseer, 2000.
[98] G. Van Horn, O. Mac Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, S. Belongie, The inaturalist species classification and detection dataset, Proceedings of the IEEE conference on computer vision and pattern recognition (2018) 8769–8778.
[99] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, Y. Bengio, Manifold mixup - better representations by interpolating hidden states, in: International Conference on Machine Learning (ICML), 2019.
[100] Vilfredo, P., Cours d'economie politique 6, 549–552. doi: 10.1086/250536.
[101] Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S., 2011a. The caltech-ucsd birds-200-2011 dataset.
[102] Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S., 2011b. The caltech-ucsd birds-200-2011 dataset.
[103] J. Wang, T. Lukasiewicz, X. Hu, J. Cai, Z. Xu, Rsg – a simple but effective module for learning imbalanced datasets, Computer Vision and Pattern Recognition (CVPR), 2021.
[104] J. Wang, W. Zhang, Y. Zang, Y. Cao, J. Pang, T. Gong, K. Chen, Z. Liu, C.C. Loy, D. Lin, Seesaw loss for long-tailed instance segmentation, Computer Vision and Pattern Recognition (CVPR), 2021.
[105] P. Wang, K. Han, X.S. Wei, L. Zhang, L. Wang, Contrastive learning based hybrid networks for long-tailed image classification, Computer Vision and Pattern Recognition (CVPR), 2021.
[106] Wang, T., Li, Y., Kang, B., Li, J., Liew, J.H., Tang, S., Hoi, S., Feng, J., 2019. Classification calibration for long-tail instance segmentation. arXiv preprint arXiv:1910.13081.
[107] X. Wang, L. Lian, Z. Miao, Z. Liu, S.X. Yu, Long-tailed recognition by routing diverse distribution-aware experts, in: International Conference on Learning Representations (ICLR), 2021.
[108] Y. Wang, Q. Yao, J.T. Kwok, L.M. Ni, Generalizing from a few examples: A survey on few-shot learning, ACM Computing Surveys (CSUR) 53 (2020) 1–34.
[109] Y.X. Wang, D. Ramanan, M. Hebert, Learning to model the tail, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017, pp. 7032–7042.
[110] C. Wei, K. Sohn, C. Mellina, A.L. Yuille, F. Yang, Crest – a class-rebalancing self-training framework for imbalanced semi-supervised learning, Computer Vision and Pattern Recognition (CVPR), 2021.
[111] X.S. Wei, J.H. Luo, J. Wu, Z.H. Zhou, Selective convolutional descriptor aggregation for fine-grained image retrieval, IEEE Trans. Image Process. 26 (2017) 2868–2881.
[112] X.S. Wei, Y.Z. Song, O. Mac Aodha, J. Wu, Y. Peng, J. Tang, J. Yang, S. Belongie, Fine-grained image analysis with deep learning: A survey, in: IEEE Trans. Pattern Anal. Mach. Intell., 2021.
[113] G. Wu, E.Y. Chang, Aligning boundary in kernel space for learning imbalanced dataset, Fourth IEEE International Conference on Data Mining IEEE ICDM'04 (2004) 265–272.
[114] G. Wu, E.Y. Chang, Kba: Kernel boundary alignment considering imbalanced data distribution, IEEE Trans. Knowl. Data Eng. 17 (2005) 786–795.
[115] Wu, T., Huang, Q., Liu, Z., Wang, Y., Lin, D., 2020a. Distribution-balanced loss for multi-label classification in long-tailed datasets, in: European Conference on Computer Vision (ECCV).
[116] T. Wu, Z. Liu, Q. Huang, Y. Wang, D. Lin, Adversarial robustness under long-tailed distribution, Computer Vision and Pattern Recognition (CVPR), 2021.
[117] Wu, T.Y., Morgado, P., Wang, P., Ho, C.H., Vasconcelos, N., 2020b. Solving long-tailed recognition with deep realistic taxonomic classifier, in: European Conference on Computer Vision (ECCV).
[118] L. Xiang, G. Ding, J. Han, Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification, European Conference on Computer Vision, Springer (2020) 247–263.
[119] L. Xiang, G. Ding, J. Han, Increasing oversampling diversity for long-tailed visual recognition, CAAI International Conference on Artificial Intelligence, Springer (2021) 39–50.
[120] S. Xie, R. Girshick, P. Dollár, Z. Tu, K. He, Aggregated residual transformations for deep neural networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 1492–1500.
[121] Y. Yang, Z. Xu, Rethinking the value of labels for improving class-imbalanced learning, in: Conference on Neural Information Processing Systems (NeurIPS), 2020.
[122] H. Yu, J. Ni, Y. Dan, S. Xu, Mining and integrating reliable decision rules for imbalanced cancer gene expression data sets, Tsinghua Sci. Technol. 17 (2012) 666–673.
[123] H. Zhang, M. Cissé, Y.N. Dauphin, D. Lopez-Paz, mixup - beyond empirical risk minimization, in: International Conference on Learning Representations (ICLR), 2018.
[124] Zhang, J., Liu, L., Wang, P., Shen, C., To balance or not to balance: A simple-yet-effective approach for learning with long-tailed distributions arXiv:1912.04486.
[125] N. Zhang, J. Donahue, R. Girshick, T. Darrell, Part-based r-cnns for fine-grained category detection, European conference on computer vision, Springer (2014) 834–849.
[126] S. Zhang, Z. Li, S. Yan, X. He, J. Sun, Distribution alignment: A unified framework for long-tail visual recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 2361–2370.
[127] Zhang, X., Fang, Z., Wen, Y., Li, Z., Qiao, Y., 2017. Range loss for deep face recognition with long-tailed training data, in: IEEE International Conference on Computer Vision (ICCV).
[128] Y. Zhang, X.S. Wei, B. Zhou, J. Wu, Bag of tricks for long-tailed visual recognition with deep convolutional neural networks, in: The Florida AI Research Society Conference (FLAIRS Conference), 2021.
[129] H. Zheng, J. Fu, T. Mei, J. Luo, Learning multi-attention convolutional neural network for fine-grained image recognition, in: 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 5219–5227, https://doi.org/10.1109/ICCV.2017.557.
[130] Y. Zhong, W. Deng, M. Wang, J. Hu, J. Peng, X. Tao, Y. Huang, Unequal-training for deep face recognition with long-tailed noisy data, Computer Vision and Pattern Recognition (CVPR), 2019.

[131] Z. Zhong, J. Cui, S. Liu, J. Jia, Improving calibration for long-tailed recognition, Computer Vision and Pattern Recognition (CVPR), 2021.
[132] B. Zhou, Q. Cui, X.S. Wei, Z.M. Chen, Bbn - bilateral-branch network with cumulative learning for long-tailed visual recognition, Computer Vision and Pattern Recognition (CVPR), 2020.
[133] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2921–2929.
[134] Zhu, J., Park, T., Isola, P., Efros, A.A., 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks, in: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22–29, 2017, IEEE Computer Society. pp. 2242–2251. doi: 10.1109/ICCV.2017.244.
[135] L. Zhu, Y. Yang, Inflated episodic memory with region self-attention for long-tailed visual recognition, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 4343–4352, https://doi.org/10.1109/CVPR42600.2020.00440.

Yu Fu is currently a postgraduate candidate in the Department of Computer Science, Aberystwyth University, UK, under the supervision of Prof. Jungong Han and Prof. Qiang Shen. She received the B.Eng. degree in Measuring and Control Technologies and Instruments and the M. E. degree in Control Science and Engineering from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2014 and 2017. She was a deep learning algorithm researcher in Beijing MOMO Technology Co., Ltd. from 2017 to 2020. Her research interests include machine learning and computer vision.

Liuyu Xiang received his B.Sc degree from EE, University of Science and Technology of China in 2017. He is currently a Ph.D. student in School of Software, Tsinghua University. His research interests include computer vision and machine learning.

Yumna Zahid is currently pursuing her PhD in Computer Science at Aberysthwyth University, under the supervision of Professor Jungong Han. Her field of work explores anomaly detection in the context of identifying suspicious activity in surveillance videos. She received her MS in Computer Science from the National University of Computer and Emerging Sciences (NUCES), Karachi, Pakistan and her BS in Computer and Information Sciences from Pakistan Institute of Engineering and Applied Sciences (PIEAS), Pakistan. Prior to her doctoral studies, she worked in a Computer Vision research group affiliated with Pakistan National Center for Big Data and Cloud Computing (NCBC) on projects for smart city initiatives funded by the Planning Commission of Pakistan and Higher Education Commission.

Guiguang Ding is currently an Associate Professor with the School of Software, Tsinghua University, China. Before joining the School of Software in 2006, he was a Postdoctoral Research Fellow of the Department of Automation, Tsinghua University. He has published over 100 papers in major journals and conferences, including the IEEE Transactions on Image Processing, IEEE Transactions on Multimedia, IEEE Transactions on Knowledge and Data Engineering, SIG IR, AAAI, ICML, IJCAI, CVPR, and ICCV. His current research interests include the areas of multimedia information retrieval, computer vision, and machine learning.

Tao Mei received the B.E. and Ph.D. degrees from the University of Science and Technology of China, Hefei, China, in 2001 and 2006, respectively. He is currently the Vice President of JD.COM and the Deputy Managing Director of JD AI Research, where he also serves as the Director for the Computer Vision and Multimedia Laboratory. Prior to joining JD.COM in 2018, he was a Senior Research Manager with Microsoft Research Asia, Beijing, China. He is also an Adjunct Professor with the University of Science and Technology of China, The Chinese University of Hong Kong, Shenzhen, and Ryerson University. He has authored or coauthored over 200 publications (with 12 best paper awards) in journals and conferences, ten book chapters, and edited five books. He holds over 25 U.S. and international patents. He is a fellow of IAPR in 2016, a Distinguished Scientist of ACM in 2016, and a Distinguished Industry Speaker of the IEEE Signal Processing Society in 2017. He is the General Co-Chair of IEEE ICME 2019 and the Program Co-Chair of ACM Multimedia 2018, IEEE ICME 2015, and IEEE MMSP 2015. He is or has been an Editorial Board Member of IEEE Transactions on Image Processing, IEEE Transactions on Circuits and Systems for Video Technology, IEEE Transactions on Multimedia, ACM Transactions on Multimedia Computing, Communications, and Applications, and Pattern Recognition.

Qiang Shen holds the Established Chair in Computer Science and is Pro Vice-Chancellor for Faculty of Business and Physical Sciences, Aberystwyth University. He has authored two research monographs and more than 410 peer-reviewed papers, including one receiving an Outstanding Transactions Paper Award from IEEE.

Jungong Han is currently a Full Professor and Chair in Computer Science at Aberystwyth University, UK. His research interests span the fields of video analysis, computer vision and applied machine learning. He has published over 200 papers, including 80 + IEEE Trans and 50 + A* conference papers.