Supervised Exploratory Learning for Long-Tailed Visual Recognition

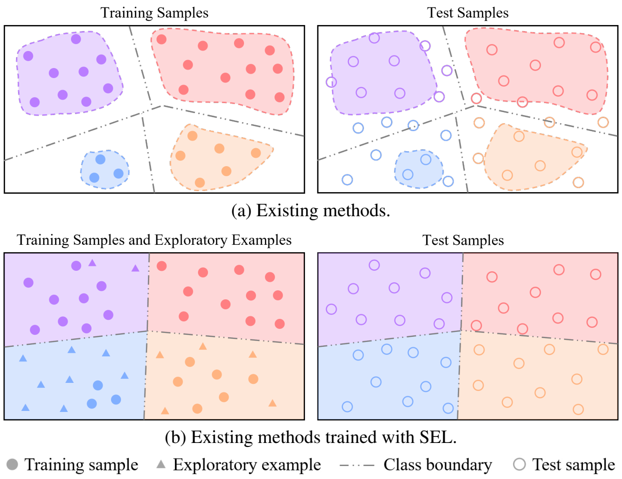
Zhongquan Jian[1,3,†], Yanhao Chen[2,†], Yanchen wang[4], Junfeng Yao[1,2,3],
Meihong Wang[3,*], Qingqiang Wu[2,3,*]
[1]School of Computer and Data Science, Minjiang University, Fuzhou, China
[2]School of Film, Xiamen University, Xiamen, China
[3]School of Informatics, Xiamen University, Xiamen, China
[4]School of Computing and Data Science, Xiamen University Malaysia, Malaysia
jianzq@mju.edu.cn, wangmh@xmu.edu.cn, wuqq@xmu.edu.cn

Abstract

*Long-tailed data poses a significant challenge for deep learning models, which tend to prioritize accurate classification of head classes while largely neglecting tail classes. Existing techniques, such as class re-balancing, logit adjustment, and data augmentation, aim to enlarge decision regions of tail classes or achieve clear decision boundaries, leaving the robustness of decision regions underconsidered. This paper proposes a simple yet effective Supervised Exploratory Learning (**SEL**) framework to achieve these goals simultaneously from space exploration perspectives. SEL employs the adaptive Optimal Foraging Algorithm (OFA) to generate diverse exploratory examples, integrating Class-biased Complement (CbC) for balanced class distribution and Fitness-weighted Sampling (FwS) for space exploration. Both theoretical analysis and empirical results demonstrate that SEL enhances class balance, sharpens decision boundaries, and strengthens decision regions. SEL is a plug-and-play training framework that can be seamlessly integrated into model training or classifier adjustment stages, making it highly adaptable and compatible with existing methods and facilitating further performance improvements. Extensive experiments on various long-tailed benchmarks demonstrate SEL's superiority.*

1. Introduction

With the success of deep learning in various Computer Vision (CV) tasks, real-world applications of automated systems have become more prevalent, such as autonomous driving [49], medical diagnosis [11], security surveillance [14], and so on. This success is largely due to the availability of large-scale labeled datasets, such as ImageNet [30], and the development of deep learning models that can learn complex patterns from balanced data [1]. However, real-world data often exhibit long-tailed distributions with heavy class imbalance, where head classes have a massive number of samples, while tail classes are associated with only a few samples [46]. Such data imbalance poses a significant challenge for developing unbiased automated recognition systems, as deep learning models tend to be biased towards head classes and disregard tail classes, which are often the most important ones in practice [28].



(a) Existing methods.

(b) Existing methods trained with SEL.

Figure 1. Comparison of decision regions and boundaries produced by (a) existing methods and (b) proposed SEL.

To solve this challenge, a multitude of studies have been proposed in recent years [46], including re-sampling [44, 47], logit adjustment [23, 38], data augmentation [27], classifier design [40], representation learning [22, 52], class-sensitive learning [13, 28], transfer learning [25, 29], ensemble learning [8, 39, 48] and decoupled training [17, 18, 41] methods. Despite their differences in principle, they all enhance the model's representation capabilities and develop an optimal classifier with clear decision boundaries to separate different classes. On the one hand, effective representation learning can alleviate the issue of imbalanced data distribution and create a balanced embedding space; on the other hand, robust classifiers can improve the recognition capability of tail classes.

Profiting from recent advanced self-supervised learning and pre-training techniques, the representation learning capability of foundational models has been significantly improved. Hence, in the pre-trained era, imbalanced data distribution primarily affects the classifier, where sufficient samples of head classes occupy decision space, while tail classes suffer from sparse, obscure decision regions. As a result, the potential of pre-trained foundational models is not fully released. During the inference stage, while informative representations of tail classes are obtained, they often differ from those in the training set. This discrepancy erodes the well-defined margins of the classifier [25], leading to the misclassification of samples from tail classes as head classes, as shown in Fig. 1(a). Recently, numerous methods have been developed to optimize the classifier. Representatively, BCL [52] and TSC [28] use Supervised Contrastive Learning (SCL) to tighten intra-class spaces and enlarge inter-class margins, significantly improving decision boundaries between classes. Furthermore, MixUp [45] and its variants [23, 42] are good at generating diversity samples that are beneficial for improving decision regions of tail classes. However, their goal of smoothing the decision space increases boundary ambiguity, which hinders classification since errors often occur at boundaries.

Inspired by space exploration of evolutionary algorithms, such as the Optimal Foraging Algorithm (OFA) [51], we attempt to address the challenges of biased decision regions and decision boundaries in long-tailed recognition from the perspective of space exploration. To this end, we propose Supervised Exploratory Learning (**SEL**), a simple yet effective framework that seamlessly integrates with existing methods to enhance their performance. SEL leverages an adaptive OFA operator to synthesize exploratory examples, compensating for insufficient tail-class data and enabling balanced decision regions and boundaries, as shown in Fig. 1(b). To be specific, SEL is composed of two modules: Class-biased Complement (CbC) and Fitness-weighted Sampling (FwS). The former is responsible for sampling more tail-class samples as target samples to balance the data distribution, while the latter selects more confident samples from adjacent classes as candidate samples to facilitate the formation of clear decision boundaries between classes. Hence, the synthesized exploratory examples are diverse and informative in representing the distribution of unseen samples in reality. SEL is a plug-and-play training framework compatible with existing long-tailed recognition methods, requiring no backbone modifications or additional parameters. The main contributions of this paper are summarized as follows:

- We attempt to address the challenge of long-tailed data from the perspective of space exploration. An adaptive OFA operator is employed to synthesize exploratory examples, which are informative and diverse to represent the distribution of unseen samples in reality.
- We propose a novel SEL framework that enables existing long-tailed recognition models to achieve balanced decision regions and clear decision boundaries, thereby improving their performance without altering network structure or increasing model parameters.
- Rational analysis proves the effectiveness of SEL in improving the classification of tail classes. Considerable performance improvements of existing long-tailed recognition methods verify the effectiveness of SEL.

2. Related Work

Data Augmentation (DA) [50] is an effective method in improving model robustness, particularly in scenarios of data scarcity or imbalance. In this section, we review recent advances in DA for long-tailed recognition, categorized into explicit and implicit methods.

2.1. Explicit Data Augmentation

Traditional DA techniques, like flipping, rotating, cropping, and padding, effectively improve robustness on long-tailed datasets and mitigate overfitting by creating new image-label pairs consistent with the training data [9, 34]. More recently, Population-Based Augmentation [10], AutoAugment [2], and Randaugment [3] have emerged to automatically choose optimal DA strategies according to characteristics of datasets, which have demonstrated their efficacy in long-tailed data [4, 21, 33]. Similarly, Wang et al. [37] generated images for tail classes using encoder variation information learned from head classes. Zada et al. [43] utilizes pure noise images as samples for tail classes.

Although generating new images is an effective data augmentation method, it is time-consuming and computationally expensive. Moreover, the diversity of the generated images is limited, as they are created through simple transformations of the original training data.

2.2. Implicit Data Augmentation

Different from explicit DA, implicit DA generates new samples by interpolating existing samples in the feature space, which can effectively improve the model's generalization ability. Representatively, MixUp [45] introduced complexity control to unexplored regions in the data space by linearly interpolating discrete sample points, effectively reducing the generalization error of the model. Subsequently, CutMix [42], an extension of MixUp, augmented data by inserting a patch from one image into another, with labels blended according to the patch areas. This method can transfer context-rich background information from dominant classes to less frequent ones by combining foreground patches from the latter with background images from the former [32]. Manifold MixUp [36] further developed this concept by applying linear interpolation to the features of input images, which proved to be more effective for long-tail learning. MetaSAug [26] learns class covariance matrices from a small, balanced validation set to augment tail classes semantically. Recently, H2T [25] improved diversity by replacing part of the tail class feature map with that of the head classes.

These findings highlight the superiority of implicit DA methods in enhancing model generalization on long-tailed data. Following this line, we propose a novel implicit DA technique, from the perspective of space exploration, to augment the diversity of training data, especially for tail classes, and improve the robustness of the long-tailed recognition model's performance.

3. Methodology

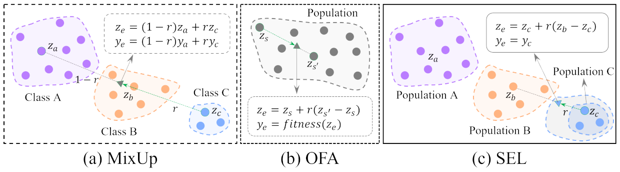
3.1. Definition

For image classification, $\{x, y\}$ represents a sample consisting of an image x and its corresponding label $y \in \mathcal{Y}$, where $\mathcal{Y} = \{1, 2, ..., C\}$ is the set of classes. In the long-tailed scenario, different classes have varying sample sizes. The imbalance ratio $\rho = \frac{n_{\max}}{n_{\min}}$ measures the skewness of the data distribution, where $n_{\max}$ and $n_{\min}$ represent the maximum and minimum number of training samples across classes. Typically, training samples for head classes are significantly greater than those for tail classes.

Existing long-tailed recognition methods usually learn a complex function mapping from the input space $\mathcal{X}$ to the target space $\mathcal{Y}$. This function is usually implemented as the composition of an encoder $f : \mathcal{X} \to \mathcal{Z} \in \mathbb{R}^d$ and a linear classifier $w : \mathcal{Z} \to \mathcal{Y}$. Both are crucial for the final classification accuracy [46].

3.2. Motivation

Firstly, the poor performance of the tail classes is primarily due to their lack of training samples. A straightforward solution is to increase the number of tail class samples through methods such as re-sampling [44] or to emphasize their importance through re-weighting [47]. Although these methods are effective, they do not generate new, diverse samples for tail classes different from the training data, limiting the enhancement of generalization. Secondly, MixUp [45] and its variants [23, 36, 42] are good at generating diverse samples that are beneficial to improving the decision regions of the tail classes. However, on the one hand, head-class samples are more likely to be randomly selected to construct mixed samples, worsening the imbalance of training data. On the other hand, their goal of smoothing the decision space intensifies the boundary ambiguity, as shown in Fig. 2(a), where the mixed sample z_e may fall into another class's region. Finally, methods like BCL [52] and TSC [28] use SCL [19] to tighten intra-class spaces and expand inter-class margins, significantly improving decision boundaries but reducing decision region robustness while increasing false positives risk.



(a) MixUp (b) OFA (c) SEL

Figure 2. The differences and relationships among MixUp, OFA and SEL. MixUp linearly interpolates two random samples (typically from different classes, such as z_a and z_c) to create a new sample, while OFA leverages the better individual ($z_{s'}$) to guide weaker individuals (z_s) toward better evolution. Combining their strengths, SEL gradually expands and reinforces minority decision regions by synthesizing adjacent exploratory examples.

To address these issues simultaneously, we propose to synthesize exploratory examples to enhance the diversity of the training data and improve the robustness of the decision regions and boundaries. As shown in Fig. 2(b), OFA [51] is good at exploring the sample space by leveraging the better individual (evaluated by the task-specific *fitness* function) to guide weaker ones, promoting more effective evolution and optimization. Combining the strengths of MixUp and OFA, SEL views samples of the same class as a population and expands its decision region by synthesizing exploratory examples in uncharted spaces, as illustrated in Fig. 2(c).

Note that MixUp is an intuitive and effective enhancement method applied at the input layer [7]. However, due to varying model structures and input patterns, unifying these methods is challenging. Consequently, most enhanced training frameworks [22, 25, 36] operate in the feature space, as most classification models first derive feature representations before classifying images. Similarly, SEL is also operated in the feature space, making it compatible with various models and methods.

3.3. Supervised Exploratory Learning

SEL is a compatible framework that can be seamlessly integrated into existing methods. In SEL, an adaptive OFA operator is employed to synthesize exploratory examples to supplement and balance the original training samples.

3.3.1. Exploratory Example Synthesization

Inspired by the space exploration process of evolutionary algorithms, we propose to synthesize exploratory examples using an adaptive OFA operator. The OFA [51], a meta-heuristic algorithm introduced in 2017, is characterized by its simple structure, strong global search efficiency, and proven affine invariance [15]. It simulates animals' foraging behavior, where each individual follows better ones to explore the environment. In OFA and its variants, the operator is defined in dimension. Here, we extend its formulation to a vector operation. Specifically, we treat samples in the batch as populations, and synthesize exploratory examples by using the following formula:

$$z_e = z_s + r(z_{s'} - z_s) \tag{1}$$

where $z_s \in \mathcal{Z}$ and $z_{s'} \in \mathcal{Z}$ denote the target and candidate samples, respectively. $r \in (0, 1)$ is a coefficient that controls the exploration distance, which is randomly sampled during model training. Generally, OFA selects the candidate sample $z_{s'}$ with higher fitness to guide weaker individuals (z_s) toward better evolution. To adapt OFA to the requirement of long-tailed recognition, we introduce two modules, CbC and FwS, to regulate the selection of target and candidate samples, respectively.

3.3.2. Class-biased Complement (CbC)

To alleviate the class imbalance problem, diverse exploratory examples are synthesized to complement the tail classes. We first count the number of samples in each class and then identify the additional samples required (*i.e.,* exploratory examples) to balance the class distribution. As illustrated in Fig. 2(c), the target sample's label is assigned to the exploratory example, eliminating the decision boundary ambiguity issue present in MixUp. Hence, synthesizing varying numbers of exploratory examples for different classes becomes a task of selecting target samples. In practice, training samples with the target class are allowed to be selected as the target samples multiple times. Specifically, for a sample z_s in the batch, it will be utilized as the target sample k_c times if it belongs to class c.

$$k_c = \mathit{Ceil}(\frac{n_{max} - n_c}{n_c}) \tag{2}$$

where n_c denotes the sample number of class c, and n_{max} represents the sample size of the largest head class. $\mathit{Ceil}(*)$ is the ceiling function.

In this way, the issue of class imbalance is addressed by synthesizing exploratory examples with different classes, ensuring that the number of samples in each class is balanced. The key challenges are guaranteeing the diversity of synthesized exploratory examples and the rationality of assigned labels, a crucial difference that distinguishes SEL from MixUp and its variants. As shown in Fig. 3, our solution is to identify neighboring classes and place exploratory examples in the regions between the target and adjacent classes, ensuring rational label assignment.

3.3.3. Neighboring Class Recognition

In MixUp and its variants, the label of the synthesized sample is determined by linear interpolation of the target and candidate samples. This approach results in label misassignment if the synthesized sample falls within the decision region of another class. To avoid this situation, we propose selecting candidate samples from neighboring classes, and placing synthesized samples between the target and adjacent class regions. Specifically, for samples in the batch $\mathcal{B}$, we calculate the center of each class and then determine the Euclidean distances between the target and the others.

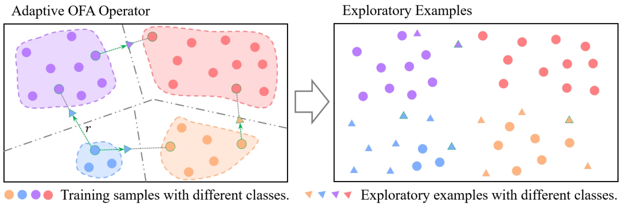

Adaptive OFA Operator
r
Exploratory Examples
Training samples with different classes.
Exploratory examples with different classes.

Figure 3. Synthesize exploratory examples by the adaptive OFA operator. More exploratory examples are generated for tail classes.

$$d(c, c') = \| \frac{1}{|\mathcal{B}^c|} \sum_{z_i \in \mathcal{B}}^{|\mathcal{B}|} z_i \cdot \mathbb{I}_{y_i = c} - \frac{1}{|\mathcal{B}^{c'}|} \sum_{z_i \in \mathcal{B}}^{|\mathcal{B}|} z_i \cdot \mathbb{I}_{y_i = c'} \|_2 \tag{3}$$

where $\mathcal{B}^c$ and $\mathcal{B}^{c'}$ are sets of samples belong to class c and c' in the batch $\mathcal{B}$, respectively. $\mathbb{I}_*$ is the indicator function, and $\| * \|_2$ denotes the L2 norm. Therefore, $d(c, c')$ denotes the distance between the centers of class c and class c'. The nearest k'_c classes are chosen as neighboring classes for class c, from which a neighbor is randomly selected, such as class c', and the candidate sample will be picked from $\mathcal{B}^{c'}$ based on their fitnesses, as done in OFA.

3.3.4. Fitness-weighted Sampling (FwS)

Following the principle of OFA, the prediction probability of the correct class is considered the fitness of the sample, and the sample with higher fitness is selected as the candidate sample. To reduce the operation complexity and increase the exploration randomness, we employ probability-based sampling to select the candidate sample. Specifically, the sampling probabilities are obtained by:

$$p_{s'} = \mathit{Norm}(\{\hat{y}_i^{c'}\}_{i=1}^{|\mathcal{B}^{c'}|}) \tag{4}$$

where c' represents the selected neighboring class, $\mathcal{B}^{c'}$ is the set of samples belong to class c' in the batch $\mathcal{B}$. $\hat{y}_i^{c'}$ is the predicted probability of the i-th sample belonging to its correct class c', and $\mathit{Norm}(*)$ is the normalization function that ensure the sum of $p_{s'}$ equals 1.

In this way, for each target sample z_s, a candidate sample $z_{s'}$ with higher confidence is sampled from the batch, thereby achieving exploratory examples with diversity. FwS is essential for the validity of SEL, which is why we use the adaptive OFA operator to synthesize exploratory examples. The rational analysis is elaborated later.

3.3.5. Training with SEL Framework

SEL is a plug-and-play training framework that can be seamlessly integrated into the training process of the original model. The optimization objective of SEL is to correctly classify the exploratory examples, which is equivalent to maximizing the predicted probability of the correct class.

$$\mathcal{L}_{SEL} = -\frac{1}{C}\sum_{c=1}^{C}\left(\frac{1}{k_c}\sum_{i=1}^{k_c}\log(\hat{y}_i^c)\right) \tag{5}$$

where $\hat{y}_i^c$ denotes the predicted probability of the i-th exploratory example belonging to the c-th class, and k_c is the number of exploratory examples synthesized for class c.

Finally, the overall optimization objective is defined as:

$$\mathcal{L} = (1-\lambda)\mathcal{L}_{Task} + \lambda\mathcal{L}_{SEL} \tag{6}$$

where $\mathcal{L}_{Task}$ is the optimization objective of the original method, and λ is a weighting coefficient that controls the impact of SEL loss.

3.4. Rationale Analysis

The exploitation process of OFA has proved to be effective and rational [51]. In this section, we provide a rational analysis of SEL. For clarity, we denote the weights of the linear classifier as $w = \{w_c\}_{c=1}^{C}$, where C is the number of classes, and $w_c \in \mathbb{R}^{d\times 1}$ denotes the weights of class c, d is the feature dimension. For a sample $z \in \mathbb{R}^{d\times 1}$, the predicted probability of class c can be expressed as:

$$\hat{y}^c = \frac{e^{w_c^T z}}{\sum_{c'=1}^{C} e^{w_{c'}^T z}} \tag{7}$$

Hence, $\hat{y}^c \propto w_c^T z$. The classifier attempts to assign a higher probability to the target class, *i.e.*, $\hat{y}^c > \hat{y}^{c'}$, where c and c' denote the target and other classes, respectively.

Without loss of generality, the selection of the target sample (z_s) and candidate sample (z'_s) can be divided into two cases because they are picked from different classes, as summarized in Tab. 1. We use the subscripts t and h to represent their relative relation. Generally, the first case dominates, as the CbC module assigns more tail-class samples as target samples, and the long-tailed nature increases the likelihood of candidate samples belonging to head classes.

Case	z_s	$z_{s'}$	z_e	Goal
1	z_t	z_h	$z_t + r(z_h - z_t)$	$\hat{y}_e^t > \hat{y}_e^h$
2	z_h	z_t	$z_h + r(z_t - z_h)$	$\hat{y}_e^h > \hat{y}_e^t$

Table 1. Two types of synthesized exploratory examples.

For the first case, the goal of $\hat{y}_e^t > \hat{y}_e^h$ is equivalent to:

$$\begin{gathered} w_t^T z_e - w_h^T z_e > 0 \Rightarrow \\ w_t^T(z_t + r(z_h - z_t)) - w_h^T(z_t + r(z_h - z_t)) > 0 \end{gathered} \tag{8}$$

After expanding and transforming the inequality:

$$\begin{aligned} w_t^T z_t - w_h^T z_t &> \frac{r}{1-r}(w_h^T z_h - w_t^T z_h) \\ \Rightarrow \hat{y}_t^t - \hat{y}_t^h &> \frac{r}{1-r}(\hat{y}_h^h - \hat{y}_h^t) \end{aligned} \tag{9}$$

where $\hat{y}_h^h$ and $\hat{y}_h^t$ represent probabilities of the candidate sample (z_h) belonging to the head class and tail classes, respectively. On one hand, the classifier favors the head class, yielding $\hat{y}_h^h - \hat{y}_h^t > 0$. On the other hand, The FwS module ensures this by selecting high-confidence candidates. Therefore, the right item is positive, and $\hat{y}_t^t > \hat{y}_t^h$ is achieved, meaning that optimizing tail-class exploratory examples effectively improves the classification accuracy of original tail-class samples.

Similarly, for the second case, the optimization goal of SEL is to make $\hat{y}_e^h > \hat{y}_e^t$, yielding:

$$\hat{y}_h^h - \hat{y}_h^t > \frac{r}{1-r}(\hat{y}_t^t - \hat{y}_t^h) \tag{10}$$

where $\hat{y}_t^t$ and $\hat{y}_t^h$ represent probabilities of the candidate sample (z_t) belonging to the tail class and head class, respectively. Although the classifier tends to misclassify tail-class samples as head classes, the FwS mitigates this by selecting better candidates, which possess higher probabilities in the correct classes. Hence, optimizing the head-class exploratory examples does not cause significant damage to the original head sample.

4. Experiments

4.1. Dataset

CIFAR-10/100-LT [5] are the subsets of CIFAR-10/100 [20]. Both contain 50,000 training images and 10,000 validation images, each of size 32×32, with 10 and 100 classes, respectively. The imbalance factors used in the experiment are set to 100, 50, and 10.

ImageNet-LT [30] is a long-tailed version of ImageNet [6] by sampling a subset following the Pareto distribution with power value $\alpha = 0.64$. It consists of 115.8K images of 1000 classes in total, with 1280 to 5 images per class.

iNaturalist 2018 [12] is a large-scale dataset containing 437.5K images from 8,142 classes. It is long-tailed by nature, with an extremely imbalanced distribution.

4.2. Basic Setting

4.2.1. Compared Methods

To evaluate the effectiveness of our proposed SEL, we integrated it into several existing methods introduced in recent years, including re-balanced techniques (IB [31], GCL

[23], KPS [24]), contrastive learning (BCL [52]), feature compression (FCC [22]), and data augmentation strategies (H2T [25], GLMC [7]). These methods were chosen not only for their representativeness but also for their publicly available code and configurations, ensuring ease of reproducibility. Most employ a two-stage training strategy, with the first stage dedicated to representation learning and the second to classification. Additionally, we use ResNet models as baselines to directly compare the performance improvements achieved by MixUp and SEL.

4.2.2. Implementation Details

We reproduce the compared methods using their official codes and configurations, setting the random seed to 2024 for fair comparison. The base ResNet models are configured identically to those in Li et al. [23]. To simplify application, we first determine the optimal way to apply SEL on different datasets by using the base model, *i.e.*, ResNet-32 for CIFAR-10/100-LT and ResNet-50 for others, then replace it with corresponding models to assess SEL's effectiveness on these models. Specifically, for CIFAR-100-LT, SEL is applied during the final 20 training epochs. For other datasets, SEL is applied by further fine-tuning the classifier for 20 epochs on CIFAR-10-LT and 10 epochs on ImageNet-LT and iNaturalist 2018. The weight coefficient λ in Eq. (6) is set to 0.9, with the number of neighboring classes k'_c set to 2, 5, 10, and 50 for CIFAR-10-LT, CIFAR-100-LT, ImageNet-LT, and iNaturalist 2018, respectively. Top-1 accuracy serves as the evaluation metric.

4.3. Improvement of Existing Methods

4.3.1. Results on CIFAR-10/100-LT

For CIFAR-10/100-LT, the results with and without SEL are presented in Tab. 2. Across 48 experiment groups, our proposed SEL significantly improves 44 of them, with an average increase of 1.37% (4.57% max and 0.02% min). Extensive experimental results fully verify the effectiveness of our method on long-tailed recognition, particularly for scenarios with high imbalance ratios. Generally, integrating MixUp into the vanilla ResNet-32 model significantly enhances performance, highlighting the effectiveness of data augmentation. However, our proposed SEL achieves even greater improvements across all scenarios, and even surpasses existing methods. For example, ResNet-32+SEL outperforms IB on both datasets except for CIFAR-10-LT-10, and surpasses GCL, KPS, and H2T on CIFAR-100-LT, though it underperforms on CIFAR-10-LT.

Additionally, SEL significantly enhances the performance of existing methods, with particularly pronounced improvements in high-imbalance scenarios. For example, SEL is notably effective on KPS at an imbalance ratio of 100 but shows limited impact at other imbalance ratios. In addition, GLMC+SEL and FCC+SEL significantly outperform their original counterparts on both datasets, with average improvements of 2.10% and 2.14%, respectively. For the re-balanced methods IB, GCL, and KPS, SEL yields greater improvements on CIFAR-100-LT than on CIFAR-10-LT, demonstrating its superiority in scenarios with more classes. However, SEL is less effective on BCL and H2T. BCL uses contrastive learning to reinforce balanced decision regions, aligning with SEL's objective. The difference is that SEL further enhances the robustness of decision regions, resulting in additional improvements. H2T is also an evolutionary algorithm-inspired method that generates diverse samples through cross-mutation, following the principles of Genetic Algorithm (GA), thus indicating the fundamental difference between SEL and H2T stems from the distinction between OFA and GA. The slight improvement of H2T+SEL further validates SEL's effectiveness in generating diverse samples, aligning with H2T's principles.

4.3.2. Results on ImageNet-LT and iNaturalist 2018

For ImageNet-LT and iNaturalist 2018, experimental results are listed in Tab. 3 and Tab. 4, respectively. Following previous works [7, 16], also report accuracy across three class splits: Many ($n_c \geq 100$), Medium ($20 < n_c < 100$), and Few ($n_c \leq 20$). Overall, SEL outperforms MixUp on both datasets, and consistently enhances the compared methods, achieving an average improvement of 1.25% (max 3.06%, min 0.61%) on ImageNet-LT and 1.53% (max 3.15%, min 0.52%) on iNaturalist 2018. As seen, training with SEL typically improves the performance of Few classes while slightly reducing that of Many classes, aligning with the rationale proved in Sec. 3.4. However, the greater gain in Few classes outweighs the slight decline in Many classes, leading to an overall improvement in the All classes.

4.4. Further Analysis

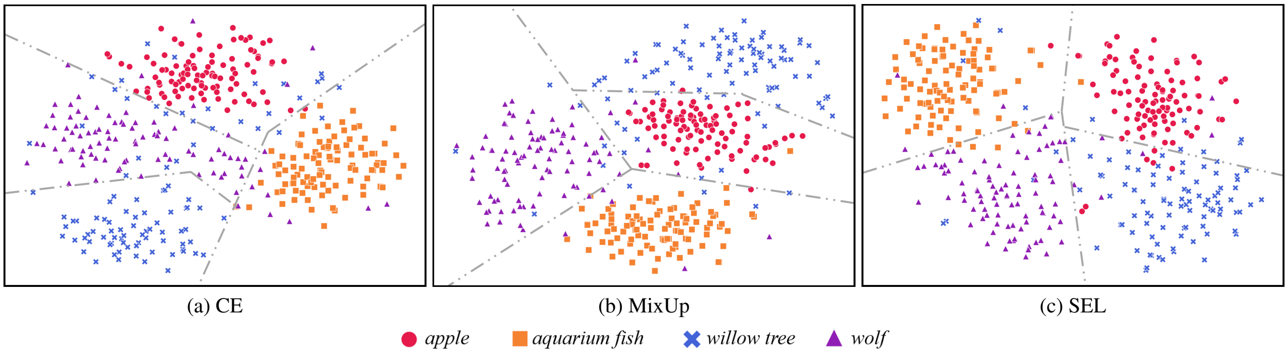
To further analyze the principles of SEL quantitatively, we conducted a series of ablation studies on the CIFAR-10/100-LT datasets using ResNet-32 [9] as the base model. More detailed results can be found in the Appendix.

4.4.1. Visualization of Decision Regions

Fig. 4 visualizes the decision regions and boundaries of different classes using t-SNE [35] across various methods. These models are trained on the CIFAR-100-LT with an imbalance ratio of 100 and tested on a balanced test set. For comparison, we selected four classes from CIFAR-100 test set for visualization: where *apple* and *aquarium fish* are the head classes with the most training samples, and *willow tree* and *wolf* are the tail classes with the fewest training samples. The decision boundaries are derived using the SVM algorithm. From Fig. 4, we can observe that SEL results in more balanced decision regions for both head and tail classes, aligning with our motivation.

Methods		CIFAR-100-LT			CIFAR-10-LT		
Avenue	Imbalance Ratio ρ	100	50	10	100	50	10
-	ResNet-32 [9]	39.96	46.01	56.66	71.96	75.95	85.92
	+ MixUp [45]	$41.55_{\uparrow 1.59}$	$47.11_{\uparrow 1.10}$	$57.18_{\uparrow 0.52}$	$73.43_{\uparrow 1.47}$	$77.19_{\uparrow 1.24}$	$86.37_{\uparrow 0.45}$
	+ SEL	$44.53_{\uparrow 4.57}$	$49.64_{\uparrow 3.85}$	$59.17_{\uparrow 2.51}$	$75.84_{\uparrow 3.88}$	$79.81_{\uparrow 3.86}$	$86.78_{\uparrow 0.86}$
ICCV 2021	IB [31]	40.50	46.31	56.75	75.11	79.95	88.01
	+ SEL	$42.34_{\uparrow 1.84}$	$47.95_{\uparrow 1.64}$	$57.57_{\uparrow 0.82}$	$75.04_{\downarrow 0.07}$	$79.81_{\downarrow 0.14}$	$88.13_{\uparrow 0.12}$
CVPR 2022	BCL [52]	51.76	56.51	67.90	83.61	86.13	90.10
	+ SEL	$52.30_{\uparrow 0.54}$	$57.25_{\uparrow 0.74}$	$68.43_{\uparrow 0.53}$	$84.44_{\uparrow 0.83}$	$86.31_{\uparrow 0.18}$	$90.26_{\uparrow 0.16}$
CVPR 2022	GCL [23]	46.50	51.72	61.79	80.56	84.74	89.65
	+ SEL	$47.89_{\uparrow 1.39}$	$52.74_{\uparrow 1.02}$	$62.41_{\uparrow 0.62}$	$81.86_{\uparrow 1.30}$	$85.13_{\uparrow 0.39}$	$89.78_{\uparrow 0.13}$
CVPR 2023	FCC+CE [22]	40.20	45.93	57.80	73.80	79.57	87.75
	+ SEL	$42.33_{\uparrow 2.13}$	$48.05_{\uparrow 2.12}$	$58.75_{\uparrow 0.95}$	$77.88_{\uparrow 4.08}$	$82.10_{\uparrow 2.53}$	$88.80_{\uparrow 1.05}$
CVPR 2023	GLMC [7]	53.91	58.87	68.07	83.68	86.90	91.16
	+ SEL	$56.48_{\uparrow 2.57}$	$61.13_{\uparrow 2.26}$	$70.75_{\uparrow 2.68}$	$85.40_{\uparrow 1.72}$	$88.57_{\uparrow 1.67}$	$92.83_{\uparrow 1.67}$
TPAMI 2023	KPS [24]	41.97	47.92	59.59	82.32	84.29	89.10
	+ SEL	$44.01_{\uparrow 2.04}$	$47.80_{\downarrow 0.12}$	$60.03_{\uparrow 0.44}$	$83.48_{\uparrow 1.16}$	$84.49_{\uparrow 0.20}$	$89.34_{\uparrow 0.24}$
AAAI 2024	H2T+CE [25]	42.27	47.58	58.24	79.91	82.80	88.77
	+ SEL	$42.45_{\uparrow 0.18}$	$47.80_{\uparrow 0.22}$	$58.66_{\uparrow 0.42}$	$80.43_{\uparrow 0.52}$	$82.80_{\downarrow 0.00}$	$88.79_{\uparrow 0.02}$

Table 2. Comparisons between raw methods and their SEL-enhanced counterparts on CIFAR-LT datasets.


(a) CE

(b) MixUp

(c) SEL

Figure 4. Visualization of decision regions and boundaries on the CIFAR-100 test set (each class contains 100 test samples), where *apple* and *aquarium fish* are head classes with the most training samples, and *willow tree* and *wolf* are tail classes with the fewest training samples.

4.4.2. Role of Key Modules

The CbC and FwS modules are the core components of SEL. To examine their roles, we conduct ablation experiments on CIFAR-100-LT with imbalance ratios of 100 and 10. Tab. 5 presented the experimental results, showing that the CbC module significantly enhances the baseline method, with improvements of 3.48% (39.96% → 43.44%) and 1.18% (56.66% → 57.84%) for imbalance ratios of 100 and 10, respectively. These results highlight the CbC module's crucial role in balancing data distribution, enabling the model to learn more effectively from tail classes. However, the FwS module has a minimal impact without the CbC module, yielding only 0.30% and 0.27% improvements for imbalance ratios of 100 and 10, respectively. Its effectiveness is fully realized when combined with the CbC module, leading to a significant boost in model performance. More ablation studies are shown in the Appendix.

4.4.3. Impact of different parameters

In this section, we conduct exploratory experiments on the impact of the number of neighboring classes (k'_c) in the FwS module. Experimental results are summarized in Fig. 5, where k'_c is varied from 1 to 10 and relative accuracy (values are adjusted by subtracting their mean) is utilized for com-

Method	Many	Medium	Few	All
ResNet-50	63.38	33.74	26.04	44.80
+ Mixup	62.97	35.44	28.58	$45.71_{\uparrow 0.91}$
+ SEL	64.14	38.41	31.27	$47.86_{\uparrow 3.06}$
BCL	65.83	53.16	36.27	55.67
+ SEL	65.45	54.82	36.53	$56.27_{\uparrow 0.60}$
GCL	65.04	52.77	35.69	55.51
+ SEL	65.13	53.94	39.18	$56.12_{\uparrow 0.61}$
FCC+CE	65.52	52.18	33.83	54.68
+ SEL	64.74	54.00	35.66	$55.60_{\uparrow 0.92}$
GLMC	69.42	52.49	30.41	56.17
+ SEL	68.67	54.37	38.28	$57.24_{\uparrow 1.07}$

Table 3. Comparisons on the ImageNet-LT dataset.

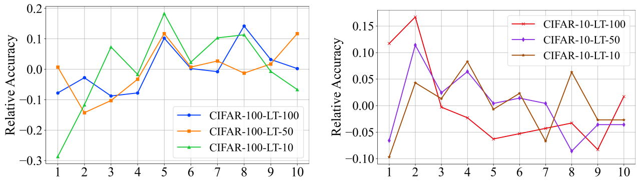
Method	Many	Medium	Few	All
ResNet-50	74.89	68.24	61.42	65.95
+ Mixup	71.34	69.53	63.88	$67.29_{\uparrow 1.34}$
+ SEL	73.04	70.38	66.74	$69.10_{\uparrow 3.15}$
BCL	73.47	71.99	68.82	70.80
+ SEL	71.99	73.04	69.21	$71.32_{\uparrow 0.52}$
GCL	73.75	72.28	68.34	70.76
+ SEL	72.63	73.58	70.05	$72.00_{\uparrow 1.24}$
FCC+CE	72.78	71.09	67.41	69.70
+ SEL	71.17	72.41	69.55	$71.09_{\uparrow 1.39}$
GLMC	76.04	73.69	71.77	73.14
+ SEL	74.68	75.14	73.66	$74.51_{\uparrow 1.37}$

Table 4. Comparisons on the iNaturalist 2018 dataset.

CbC	FwS	Top-1 Accuracy	
		ρ =100	ρ =10
×	×	39.96	56.66
×	✓	$40.26_{\uparrow 0.30}$	$56.93_{\uparrow 0.27}$
✓	×	$43.44_{\uparrow 3.48}$	$57.84_{\uparrow 1.18}$
✓	✓	$44.53_{\uparrow 4.57}$	$59.17_{\uparrow 2.51}$

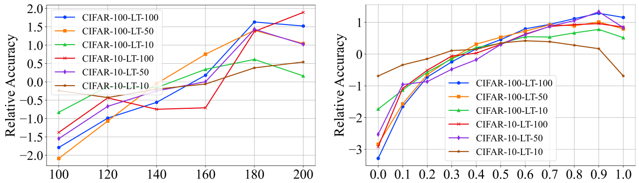
Table 5. Ablations of key modules.

parison. As seen, the model performs better with a higher $k\prime_c$ on CIFAR-100-LT but with a lower $k\prime_c$ on CIFAR-10-LT. This difference arises from the number of classes, as CIFAR-100-LT has 100 classes, whereas CIFAR-10-LT has only 10. Based on these observations, we empirically set $k\prime_c$ to 5 for CIFAR-100-LT and 2 for CIFAR-10-LT.



(a) CIFAR-100-LT (b) CIFAR-10-LT

Figure 5. Changes with different neighboring class numbers.



(a) start epochs (b) weight (λ)

Figure 6. Changes with different parameters.

4.4.4. When to Start SEL Training

In this section, we explore the optimal starting point for SEL training to maximize computational efficiency, as its synthesis of additional exploratory examples increases computational costs. Specifically, we integrate SEL into the training process at different epochs, ranging from 100 to 200, as shown in Fig. 6(a). Here, starting at 200 epochs indicates a two-stage training mode, where SEL is applied with an additional 20 epochs after the original training process. Similarly, relative accuracy is used as the ordinate for clearly showing the parameter's impact. It is obvious that the model's performance steadily improves as the SEL starting point increases, with most datasets peaking at 180 epochs, except for CIFAR-10-LT-100 and CIFAR-10-LT-10. Therefore, we set the starting point to 180 epochs for CIFAR-100-LT and 200 epochs for CIFAR-10-LT.

Building on the optimal starting points, we further examine the impact of parameter λ in Eq.(6). Fig. 6(b) illustrates the performance variation with λ ranges from 0.0 to 1.0. Note that $\lambda = 0.0$ indicates the absence of SEL, while $\lambda = 1.0$ signifies its full integration. For most datasets, as λ increases, the model's performance gradually improves, peaking at $\lambda = 0.9$ before declining. However, for CIFAR-10-LT-10, the simplest dataset, performance remains relatively stable, with a peak at $\lambda = 0.6$.

5. Conclusion

In this paper, we propose SEL, a simple yet effective training framework that addresses long-tailed challenges through the space exploration perspective. SEL is a plug-and-play method that can seamlessly integrate with existing methods, enhancing performance without altering model structure or increasing parameters. In SEL, an adaptive

OFA operator is used to synthesize exploratory examples, particularly for tail classes, to create a balanced data distribution. These diverse examples effectively capture the characteristics of unseen tail classes, enabling the model to learn more robust decision regions and boundaries. The effectiveness of SEL is validated mathematically through rational analysis and empirically through experiments on various long-tailed datasets and methods. Experimental results and further analysis demonstrate SEL's significance in enhancing existing methods.

References

[1] Yoshua Bengio, Yann Lecun, and Geoffrey Hinton. Deep learning for ai. *Communications of the ACM*, 64(7):58–65, 2021. 1

[2] Ekin D. Cubuk, Barret Zoph, Dandelion Mané, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation strategies from data. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 113–123, 2019. 2

[3] Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Advances in Neural Information Processing Systems*, pages 18613–18624. Curran Associates, Inc., 2020. 2

[4] Jiequan Cui, Zhisheng Zhong, Shu Liu, Bei Yu, and Jiaya Jia. Parametric contrastive learning. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 695–704, 2021. 2

[5] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge J. Belongie. Class-balanced loss based on effective number of samples. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9268–9277, 2019. 5

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 5

[7] Fei Du, Peng Yang, Qi Jia, Fengtao Nan, Xiaoting Chen, and Yun Yang. Global and local mixture consistency cumulative learning for long-tailed visual recognitions. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15814–15823, 2023. 3, 6, 7

[8] Hao Guo and Song Wang. Long-tailed multi-label visual recognition by collaborative training on uniform and rebalanced samplings. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 15089–15098, 2021. 1

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2, 6, 7

[10] Daniel Ho, Eric Liang, Xi Chen, Ion Stoica, and Pieter Abbeel. Population based augmentation: Efficient learning of augmentation policy schedules. In *Proceedings of the 36th International Conference on Machine Learning*, pages 2731–2741, 2019. 2

[11] Gregory Holste, Yiliang Zhou, Song Wang, Ajay Jaiswal, Mingquan Lin, Sherry Zhuge, Yuzhe Yang, Dongkyun Kim, Trong-Hieu Nguyen-Mau, Minh-Triet Tran, Jaehyup Jeong, Wongi Park, Jongbin Ryu, Feng Hong, Arsh Verma, Yosuke Yamagishi, Changhyun Kim, Hyeryeong Seo, Myungjoo Kang, Leo Anthony Celi, Zhiyong Lu, Ronald M. Summers, George Shih, Zhangyang Wang, and Yifan Peng. Towards long-tailed, multi-label disease classification from chest x-ray: Overview of the cxr-lt challenge. *Medical Image Analysis*, 97:103224, 2024. 1

[12] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alexander Shepard, Hartwig Adam, Pietro Perona, and Serge J. Belongie. The inaturalist species classification and detection dataset. In *2018 IEEE Conference on Computer Vision and Pattern Recognition*, pages 8769–8778, 2018. 5

[13] Chengkai Hou, Jieyu Zhang, Haonan Wang, and Tianyi Zhou. Subclass-balancing contrastive learning for long-tailed recognition. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5372–5384, 2023. 1

[14] Shuyan Hu, Wei Ni, Xin Wang, and Abbas Jamalipour. Disguised tailing and video surveillance with solar-powered fixed-wing unmanned aerial vehicle. *IEEE Transactions on Vehicular Technology*, 71(5):5507–5518, 2022. 1

[15] ZhongQuan Jian and GuangYu Zhu. Affine invariance of *meta*-heuristic algorithms. *Information Sciences*, 576:37–53, 2021. 4

[16] Yan Jin, Mengke Li, Yang Lu, Yiu-ming Cheung, and Hanzi Wang. Long-tailed visual recognition via self-heterogeneous integration with knowledge excavation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23695–23704, 2023. 6

[17] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *8th International Conference on Learning Representations*, 2020. 1

[18] Bingyi Kang, Yu Li, Sa Xie, Zehuan Yuan, and Jiashi Feng. Exploring balanced feature spaces for representation learning. In *9th International Conference on Learning Representations*, 2021. 1

[19] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673, 2020. 3

[20] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 5

[21] Jun Li, Zichang Tan, Jun Wan, Zhen Lei, and Guodong Guo. Nested collaborative learning for long-tailed visual recognition. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6939–6948, 2022. 2

[22] Jian Li, Ziyao Meng, Daqian Shi, Rui Song, Xiaolei Diao, Jingwen Wang, and Hao Xu. FCC: feature clusters compression for long-tailed visual recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24080–24089, 2023. 1, 3, 6, 7

[23] Mengke Li, Yiu-Ming Cheung, and Yang Lu. Long-tailed visual recognition via gaussian clouded logit adjustment. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6919–6928, 2022. 1, 2, 3, 6, 7

[24] Mengke Li, Yiu-Ming Cheung, and Zhikai Hu. Key point sensitive loss for long-tailed visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4): 4812–4825, 2023. 6, 7

[25] Mengke Li, Zhikai Hu, Yang Lu, Weichao Lan, Yiu-ming Cheung, and Hui Huang. Feature fusion from head to tail for long-tailed visual recognition. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, Fourteenth Symposium on Educational Advances in Artificial Intelligence*, pages 13581–13589, 2024. 1, 2, 3, 6, 7

[26] Shuang Li, Kaixiong Gong, Chi Harold Liu, Yulin Wang, Feng Qiao, and Xinjing Cheng. Metasaug: Meta semantic augmentation for long-tailed visual recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5212–5221, 2021. 3

[27] Shuang Li, Kaixiong Gong, Chi Harold Liu, Yulin Wang, Feng Qiao, and Xinjing Cheng. Metasaug: Meta semantic augmentation for long-tailed visual recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5212–5221, 2021. 1

[28] Tianhong Li, Peng Cao, Yuan Yuan, Lijie Fan, Yuzhe Yang, Rogerio Feris, Piotr Indyk, and Dina Katabi. Targeted supervised contrastive learning for long-tailed recognition. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6908–6918, 2022. 1, 2, 3

[29] Bo Liu, Haoxiang Li, Hao Kang, Gang Hua, and Nuno Vasconcelos. Gistnet: a geometric structure transfer network for long-tailed recognition. In *2021 IEEE/CVF International Conference on Computer Vision*, pages 8189–8198, 2021. 1

[30] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X. Yu. Large-scale long-tailed recognition in an open world. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2537–2546, 2019. 1, 5

[31] Seulki Park, Jongin Lim, Younghan Jeon, and Jin Young Choi. Influence-balanced loss for imbalanced visual classification. In *2021 IEEE/CVF International Conference on Computer Vision*, pages 715–724, 2021. 5, 7

[32] Seulki Park, Youngkyu Hong, Byeongho Heo, Sangdoo Yun, and Jin Young Choi. The majority can help the minority: Context-rich minority oversampling for long-tailed classification. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6877–6886, 2022. 3

[33] Jiawei Ren, Cunjun Yu, Shunan Sheng, Xiao Ma, Haiyu Zhao, Shuai Yi, and Hongsheng Li. Balanced meta-softmax for long-tailed visual recognition. In *Advances in Neural Information Processing Systems 33*, 2020. 2

[34] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015. 2

[35] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9 (11), 2008. 6

[36] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. In *Proceedings of the 36th International Conference on Machine Learning*, pages 6438–6447, 2019. 3

[37] Jianfeng Wang, Thomas Lukasiewicz, Xiaolin Hu, Jianfei Cai, and Zhenghua Xu. Rsg: A simple but effective module for learning imbalanced datasets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3784–3793, 2021. 2

[38] Peng Wang, Kai Han, Xiu-Shen Wei, Lei Zhang, and Lei Wang. Contrastive learning based hybrid networks for long-tailed image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 943–952, 2021. 1

[39] Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella X. Yu. Long-tailed recognition by routing diverse distribution-aware experts. In *9th International Conference on Learning Representations*, 2021. 1

[40] Tong Wu, Ziwei Liu, Qingqiu Huang, Yu Wang, and Dahua Lin. Adversarial robustness under long-tailed distribution. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8659–8668, 2021. 1

[41] Shiyu Xuan and Shiliang Zhang. Decoupled contrastive learning for long-tailed recognition. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, Fourteenth Symposium on Educational Advances in Artificial Intelligence*, pages 6396–6403, 2024. 1

[42] Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Seong Joon Oh, Youngjoon Yoo, and Junsuk Choe. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6022–6031, 2019. 2, 3

[43] Shiran Zada, Itay Benou, and Michal Irani. Pure noise to the rescue of insufficient data: Improving imbalanced classification by training on random noise images. In *International Conference on Machine Learning*, pages 25817–25833, 2022. 2

[44] Yuhang Zang, Chen Huang, and Chen Change Loy. Fasa: Feature augmentation and sampling adaptation for long-tailed instance segmentation. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3437–3446, 2021. 1, 3

[45] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *6th International Conference on Learning Representations*, 2018. 2, 3, 7

[46] Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, and Jiashi Feng. Deep long-tailed learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9):10795–10816, 2023. 1, 3

[47] Zizhao Zhang and Tomas Pfister. Learning fast sample reweighting without reward data. In *2021 IEEE/CVF Inter-*

national Conference on Computer Vision, pages 705–714, 2021. 1, 3

[48] Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen. BBN: bilateral-branch network with cumulative learning for long-tailed visual recognition. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9716–9725, 2020. 1

[49] Weitao Zhou, Zhong Cao, Nanshan Deng, Xiaoyu Liu, Kun Jiang, and Diange Yang. Dynamically conservative self-driving planner for long-tail cases. *IEEE Transactions on Intelligent Transportation Systems*, 24(3):3476–3488, 2023. 1

[50] Chenguang Zhu, Yichong Xu, Xiang Ren, Bill Yuchen Lin, Meng Jiang, and Wenhao Yu. Knowledge-augmented methods for natural language processing. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, pages 1228–1231, 2023. 2

[51] Guang-Yu Zhu and Wei-Bo Zhang. Optimal foraging algorithm for global optimization. *Applied Soft Computing*, 51: 294–313, 2017. 2, 3, 5

[52] Jianggang Zhu, Zheng Wang, Jingjing Chen, Yi-Ping Phoebe Chen, and Yu-Gang Jiang. Balanced contrastive learning for long-tailed visual recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6898–6907, 2022. 1, 2, 3, 6, 7