Appendix
Kill Two Birds with One Stone: Rethinking Data Augmentation for Deep Long-tailed Learning

The content of the **Appendix** is summarized as follows:

1) in Sec. A, we state the proofs of Theorem 2 (Sec. 2.1), Theorem 3 (Sec. 2.2), and Theorem 4 (Sec. 3.2).
2) in Sec. B, we summarize existing long-tailed learning (LTL) and data augmentation (DA) methods and explicitly illustrate the novelty of DODA.
3) in Sec. C, we demonstrate the details of datasets and baselines we use in experiments of DODA.
4) in Sec. D, we illustrate more detailed empirical results and analyses of DODA.

A DETAILED PROOFS

A.1 PROOF OF THEOREM 2

Proof. In Sec. 2.1, Theorem 1 states that when we approximate the ideal model f^* by minimizing the training loss (i.e., 0 training error), the latter tends to zero, while the former is greater than zero due to the augmented samples deviate from the level set of f^*. Therefore, the trained model f_θ will be biased compared to the ideal model f^*.

$$\sum_{(x,y)\in\mathcal{D}} \mathbb{E}[||y - f^*(\mathcal{O}(x))||_2^2] > 0 \wedge \sum_{(x,y)\in\mathcal{D}} \mathbb{E}[||y - f_\theta(\mathcal{O}(x))||_2^2] = 0 \Longrightarrow \text{bias} \tag{10}$$

From the perspective of the dataset, the reason for the bias is that the semantic information of some samples does not match their original labels after DA, which means that DA cannot guarantee that it is label-preserving. We reconsider this problem from the perspective of long-tailed learning. First of all, for the whole dataset $\mathcal{D}$, minimizing the training loss essentially means that the trained model f_θ should achieve 0 training error on each class, i.e.,

$$\sum_{(x,y)\in\mathcal{D}} \mathbb{E}[||y - f_\theta(\mathcal{O}(x))||_2^2] = 0 \Longrightarrow \sum_{(x,y)\in\mathcal{D}|y=c} \mathbb{E}[||y - f_\theta(\mathcal{O}(x))||_2^2] = 0 \quad \forall c \in C \tag{11}$$

Similarly, for class c and augmented samples $\{(\mathcal{O}(x), y)|y = c, (\mathcal{O}(x), y) \in \mathcal{D}\}$, when we achieve or approximately achieve the minimization of the training loss on class c, DA inevitably makes some samples of class c mismatch with their original labels, i.e., the augmented samples deviate from the level set of the ideal model f^*. Therefore, when we use the augmented samples of class c as inputs to f^*, f^* cannot predict the labels completely correctly.

$$\sum_{(x,y)\in\mathcal{D}} \mathbb{E}[||y - f^*(\mathcal{O}(x))||_2^2] > 0 \Longrightarrow \sum_{(x,y)\in\mathcal{D}|y=c} \mathbb{E}[||y - f^*(\mathcal{O}(x))||_2^2] > 0 \quad \forall c \in C \tag{12}$$

Although we achieve a seemingly optimal (ideal) training model f_θ, its fitting process on class c has actually deviated from the ideal optimization process. Therefore, the deviation between the trained model f_θ and the ideal model f^* on class c is inevitable, i.e., f_θ has class-wise bias.

$$\sum_{(x,y)\in\mathcal{D}|y=c} \mathbb{E}[||y - f^*(\mathcal{O}(x))||_2^2] > 0 \wedge \sum_{(x,y)\in\mathcal{D}|y=c} \mathbb{E}[||y - f_\theta(\mathcal{O}(x))||_2^2] = 0 \Longrightarrow \text{class-wise bias} \tag{13}$$

A.2 PROOF OF THEOREM 3

Proof. According Definition 1, the distribution of each class in the training set can be approximate as a circle in a two-dimensional feature space, and the distribution center of class c can be defined as $(\mathbb{X}_c, \mathbb{Y}_c)$ and the distribution radius is $\mathbb{R}_c$. So, the distribution span $\mathbb{S}_c$ can be expressed as follows:

$$\mathbb{S}_c \Rightarrow (\mathbb{X} - \mathbb{X}_c)^2 - (\mathbb{Y} - \mathbb{Y}_c)^2 = \mathbb{R}_c{}^2 \tag{14}$$

For the data distribution $\mathbb{S}_{c_h}$ and $\mathbb{S}_{c_t}$ of head class c_h and tail class c_t, the new distribution span after using uniform DA $\mathcal{O}(\cdot)$ can be defined as $\bar{\mathbb{S}}_{c_h} \Rightarrow (\mathbb{R}_{c_h} + \Delta_{c_h})^2$ and $\bar{\mathbb{S}}_{c_t} \Rightarrow (\mathbb{R}_{c_t} + \Delta_{c_t})^2$, and Δ_{c_h} and Δ_{c_t} represent the increase in distribution radius within each class after DA. For the same augmentation method, $\Delta_{c_h} = \Delta_{c_t}$.

Here, we define the original data distributions of head class c_h and tail class c_t as the base spaces ${\mathbb{R}_{c_t}}^2$ and ${\mathbb{R}_{c_h}}^2$, and define the expanded data distributions of head class c_h and tail class c_t as the marginal spaces $(\mathbb{R}_{c_t} + \Delta_{c_t})^2 - {\mathbb{R}_{c_t}}^2$ and $(\mathbb{R}_{c_h} + \Delta_{c_h})^2 - {\mathbb{R}_{c_h}}^2$

Then, the augmentation sensitivity of head class c_h and tail class c_t can be defined as ψ_{c_h} and ψ_{c_t}.

$$\psi_{c_h} = \frac{(\mathbb{R}_{c_h} + \Delta_{c_h})^2 - {\mathbb{R}_{c_h}}^2}{{\mathbb{R}_{c_h}}^2} \tag{15}$$

$$\psi_{c_t} = \frac{(\mathbb{R}_{c_t} + \Delta_{c_t})^2 - {\mathbb{R}_{c_t}}^2}{{\mathbb{R}_{c_t}}^2} \tag{16}$$

Therefore, we can measure the augmentation sensitivity difference between head class c_h and tail class c_t, i.e.,

$$\begin{aligned}
\psi_{c_t} - \psi_{c_h} &= \frac{(\mathbb{R}_{c_t} + \Delta_{c_t})^2 - {\mathbb{R}_{c_t}}^2}{{\mathbb{R}_{c_t}}^2} - \frac{(\mathbb{R}_{c_h} + \Delta_{c_h})^2 - {\mathbb{R}_{c_h}}^2}{{\mathbb{R}_{c_h}}^2} \\
&= \frac{{\mathbb{R}_{c_t}}^2 + 2\mathbb{R}_{c_t}\Delta_{c_t} + {\Delta_{c_t}}^2 - {\mathbb{R}_{c_t}}^2}{{\mathbb{R}_{c_t}}^2} - \frac{{\mathbb{R}_{c_h}}^2 + 2\mathbb{R}_{c_h}\Delta_{c_h} + {\Delta_{c_h}}^2 - {\mathbb{R}_{c_h}}^2}{{\mathbb{R}_{c_h}}^2} \\
&= \frac{2\mathbb{R}_{c_t}\Delta_{c_t} + {\Delta_{c_t}}^2}{{\mathbb{R}_{c_t}}^2} - \frac{2\mathbb{R}_{c_h}\Delta_{c_h} + {\Delta_{c_h}}^2}{{\mathbb{R}_{c_h}}^2} \\
&= \frac{2\mathbb{R}_{c_t}{\mathbb{R}_{c_h}}^2\Delta_{c_t} + {\mathbb{R}_{c_h}}^2{\Delta_{c_t}}^2 - 2\mathbb{R}_{c_t}^2\mathbb{R}_{c_h}\Delta_{c_h} - {\mathbb{R}_{c_t}}^2{\Delta_{c_h}}^2}{{\mathbb{R}_{c_t}}^2{\mathbb{R}_{c_h}}^2} \\
&= \frac{2\mathbb{R}_{c_t}\mathbb{R}_{c_h}(\mathbb{R}_{c_h}\Delta_{c_t} - \mathbb{R}_{c_t}\Delta_{c_h})}{{\mathbb{R}_{c_t}}^2{\mathbb{R}_{c_h}}^2} + \frac{{\mathbb{R}_{c_h}}^2{\Delta_{c_t}}^2 - {\mathbb{R}_{c_t}}^2{\Delta_{c_h}}^2}{{\mathbb{R}_{c_t}}^2{\mathbb{R}_{c_h}}^2} \\
&= 2\Delta_{c_h} \cdot \frac{\mathbb{R}_{c_h} - \mathbb{R}_{c_t}}{\mathbb{R}_{c_h}\mathbb{R}_{c_t}} + {\Delta_{c_h}}^2 \cdot \frac{{\mathbb{R}_{c_h}}^2 - {\mathbb{R}_{c_t}}^2}{{\mathbb{R}_{c_h}}^2{\mathbb{R}_{c_t}}^2} \\
&> 0
\end{aligned} \tag{17}$$

The above derivation indicating that tail classes are more sensitive to the marginal space.

For high-dimensional feature space,

Proof. Assuming that the dimension of high-dimensional features is n,the distribution center of class c is defined as $(\mathbb{X}_c^1, \mathbb{X}_c^2, ..., \mathbb{X}_c^n)$ and the distribution radius is $\mathbb{R}_c$. The distribution span $\mathbb{S}_c$ can be expressed as

$$(\mathbb{X}^1 - \mathbb{X}_c^1)^2 - (\mathbb{X}^2 - \mathbb{X}_c^2)^2 - ... - (\mathbb{X}^n - \mathbb{X}_c^n)^2 = {\mathbb{R}_c}^2 \tag{18}$$

We assume that the data distribution of head class c_h and tail class c_t are $\mathbb{S}_{c_h}$ and $\mathbb{S}_{c_t}$ and the distribution span after DA $\bar{\mathbb{S}}_{c_h}$ and $\bar{\mathbb{S}}_{c_t}$. So the augmentation sensitivity ψ of class c can be expressed as follows:

$$\psi_c = \frac{\frac{\pi^{n/2}(\mathbb{R}_c + \Delta)}{\Gamma(1+n/2)} - \frac{\pi^{n/2}\mathbb{R}_c}{\Gamma(1+n/2)}}{\frac{\pi^{n/2}\mathbb{R}_c}{\Gamma(1+n/2)}}, \tag{19}$$

and further deduce:

$$\psi_{c_t} - \psi_{c_h} = \frac{\pi^{n/2}}{\Gamma(1+n/2)} \frac{(\mathbb{R}_{c_t}\mathbb{R}_{c_h} + \Delta\mathbb{R}_{c_h})^n - (\mathbb{R}_{c_t}\mathbb{R}_{c_h} + \Delta\mathbb{R}_{c_t})^n}{{\mathbb{R}_{c_t}}^n{\mathbb{R}_{c_h}}^n} > 0 \tag{20}$$

This indicates that this theoretical explanation is equally applicable to higher-dimensional spaces.

A.3 PROOF OF THEOREM 4

Proof. We want to show that for a more dominant DA $\mathcal{O}^{k_1}$, the bias of $\mathcal{O}^{k_1}$ from P is smaller than that of $\mathcal{O}^{k_2}$ from P, where Q^{k_1} and Q^{k_2} are the level-sets of the models $f_\theta^{k_1}$ and $f_\theta^{k_2}$ learned using DA $\mathcal{O}^{k_1}$ and DA $\mathcal{O}^{k_2}$, respectively, and P is the level-set of the model f_θ trained on the original dataset.

Herr, we use Chebyshev's inequality to bound the probability that a random variable deviates from its expected value by a certain amount. Let X be a random variable that represents the deviation of $f_\theta(x)$ from y for a sample (x, y) in the original dataset. Let Y be a random variable that represents the deviation of $f_\theta^k(\mathcal{O}^k(x))$ from y for a sample (x, y) in the augmented dataset using DA $\mathcal{O}^k$. Then we have:

$$P(|X - E(X)| > t) \leq Var(X)/t^2 \tag{21}$$

$$P(|Y - E(Y)| > t) \leq Var(Y)/t^2 \tag{22}$$

The level-set bias $\delta(Q^k, P)$ can be defined as the degree of distributional deviation between the level-sets Q^k and P. Intuitively, this can be measured by the difference between $E(Y)$ and $E(X)$, or the difference between $Var(Y)$ and $Var(X)$. We assume that $E(X) = 0$, since f_θ is trained to minimize the training loss on the original dataset. Then we have:

$$\delta(Q^k, P) = |E(Y)| + |Var(Y) - Var(X)| \tag{23}$$

Now, suppose that $\mathcal{O}^{k_1}$ dominates $\mathcal{O}^{k_2}$ on class c, i.e., $\nabla^{pos}_{z_c^{k_1}} > \nabla^{pos}_{z_c^{k_2}}$. This means that $f_\theta(\mathcal{O}^{k_1}(x))$ is more likely to be equal to y than $f_\theta(\mathcal{O}^{k_2}(x))$ for samples (x, y) in class c. Therefore, we have:

$$E(Y|y = c, \mathcal{O}^{k_1}) < E(Y|y = c, \mathcal{O}^{k_2}) \tag{24}$$

$$Var(Y|y = c, \mathcal{O}^{k_1}) < Var(Y|y = c, \mathcal{O}^{k_2}) \tag{25}$$

By taking the weighted average over all classes, we obtain,

$$E(Y|\mathcal{O}^{k_1}) < E(Y|\mathcal{O}^{k_2}) \tag{26}$$

$$Var(Y|\mathcal{O}^{k_1}) < Var(Y|\mathcal{O}^{k_2}) \tag{27}$$

Hence, we conclude that:

$$\delta(Q^{k_1}, P) < \delta(Q^{k_2}, P) \tag{28}$$

This completes the proof.

B RELATED WORK

B.1 LONG-TAILED LEARNING (LTL)

Real-world training datasets typically exhibit a long-tailed class distribution, where a small fraction of classes have massive samples and the rest classes are associated with only a few samples. Unfortunately, the deep models trained by the common practice of empirical risk minimization cannot handle this distribution, resulting in a significant decrease in model performance Zhang et al. (2021b). Recently, missive novel longt-tailed learning methods have been proposed to learn a more generalized model from imbalanced training datasets, which can be divide into three main categories: class re-balancing Kang et al. (2020); Ren et al. (2020); Wang et al. (2020); Lin et al. (2017); Cui et al. (2019); Tan et al. (2020), module improvement Zhang et al. (2017b); Ouyang et al. (2016); Tang et al. (2020); Kang et al. (2020); Zhou et al. (2020); Zhang et al. (2022), and information augmentation Chu et al. (2020); Kim et al. (2020b); Hu et al. (2020); Zang et al. (2021); Park et al. (2022); Ahn et al. (2023).

Class re-balancing is the most typical strategy, which balances inter-class sample numbers or weights by re-sampling or cost-sensitive learning. On the one hand, traditional re-sampling methods, e.g., random over-sampling (ROS) and random under-sampling (RUS), achieve re-balancing by repeating the samples from tail classes and discarding the samples from head classes, but they tend to overfit to tail classes when datasets are extremely unbalanced. To this end, recent studies propose class-balanced re-sampling strategies, e.g., bi-level class-balanced sampling Wang et al. (2020) and meta learning based sampling Ren et al. (2020). Besides from the perspective of classes, scheme-oriented sampling strategies try to re-balance classes by designing some specific learning schemes, such as quintuplet sampling Huang et al. (2016) and replay based sampling Kim et al. (2020a). On the other hand, some studies, called cost-sensitive learning, re-balance classes by adjusting the loss values of different classes. For example, CB Cui et al. (2019) proposed a effective number to approximate the expected sample number of each class, and Focal loss Lin et al. (2017) used the prediction probabilities to inversely re-weight classes.

In addition to class re-balancing, researchers also explored enhancing model performance by improving network modules. A intuitive method is decoupled training, which decouples the learning procedure into representation learning and classifier training. As a pioneering work, Decoupling Kang et al. (2020) proposed a two-stage training scheme and showed some refreshing observations. KCL Kang et al. (2021) and FRS Wang et al. (2023) believed that a balanced feature space is beneficial to LTL, so they designed contrastive learning based losses to learn a more class-balanced and class-discriminative feature space. Furthermore, as a classic theory, ensemble learning is also applied to LTL by designing and combining multiple expert networks. For instance, BBN Zhou et al. (2020) proposed to use two network branches to handle LTL. Following BBN, BAGS Li et al. (2020b) explored a multi-head scheme. Not restricted to a balanced test set, SADE Zhang et al. (2022) explored the multi-expert scheme to handle test distribution-agnostic LTL.

Although the overall performance is improved, these methods cannot essentially handle the issue of lacking information, particularly on tail classes due to limited data amount. Orthogonally, some information augmentation studies seek to introduce additional information into model training, such as FTL Yin et al. (2019) and M2m Kim et al. (2020b) transferred the knowledge from head classes to enhance model training on tail classes considering the inter-class knowledge imbalance. To solve information restrictions in essence, another line of research is to apply representation augmentation or data augmentation to LTL. For example, CMO Park et al. (2022) augmented diversified minority samples by leveraging the rich context of the majority classes as background images. Considering fairness, FSR Wang et al. (2023) and CUDA Ahn et al. (2023) advocate to find appropriate augmentation strength for each class. However, *although these methods enrich the overall information to a certain extent and improve model performance, they ignored the sacrifice of some classes behind this improvement*. **For this reason, we jointly pay attention to the inherent data-wise imbalance and extrinsic augmentation-wise imbalance, thereby minimizing the sacrifice.**

B.2 DATA AUGMENTATION

DA has been applied in many fields because it can effectively alleviate overfitting and improve model generalization performance. DA is simple in design, and various DAs can be achieved through image manipulation, e.g., filp, crop, and rotate Robbins & Monro (1951). Recently, mixup based DA methods are proposed to improve model robustness by fusing two images and their labels Zhang et al. (2017a); Tokozume et al. (2018). Considering the diversity of DA, some studies try to combine them randomly or in order, such as AutoAugment Cubuk et al. (2019), Fast AutoAugment Lim et al. (2019), DADA Li et al. (2020a), and RandAugment Cubuk et al. (2020). In addition, researchers are improving DAs to make them suitable for LTL, however, they ignore that DA is class-independent, and thus may cause a mismatch between augmented data and actual labels Park et al. (2022); Wang et al. (2023); Ahn et al. (2023). Therefore, **it is necessary to design a class-dependent long-tailed DA to allow each class to choose an appropriate augmentation method.**

C DATASET AND BASELINE DETAILS

C.1 DATASET

CIFAR-100-LT Cao et al. (2019): is a long-tailed version of artificially truncated from the original balanced dataset CIFAR-100, which includes 100 different categories, 50,000 training images and 10,000 test images. The 100 categories in CIFAR-100 form 20 superclasses, each with 5 classes. CIFAR-100-LT has three imbalance ratio settings 10, 50, 100, where the imbalance ratio ρ is defined as the ratio of the sample sizes of the most frequent and least frequent classes, i.e., $\rho = N_{max}/N_{min}$.

Table 3: Statistics of the long-tailed datasets.

Dataset	# of Classes	# of Training set	# of Test set	Imbalance ratio
CIFAR-100-LT	100	50,000	10,000	{10, 50, 100}
ImageNet-LT	1,000	115,846	50,000	256
iNaturalist 2018	8,142	437,513	24,426	500

ImageNet-LT Liu et al. (2019): is a long-tailed version of artificially truncated from the original balanced dataset ImageNet, which includes 1,000 different categories, 115,846 training images and 50,000 test images. The most frequent or least frequent class has 1,280 or 5 images, so the imbalance ratio $\rho = 256$.

iNaturalist 2018 Van Horn et al. (2018): is a real-world, naturally long-tailed dataset, which includes 8,142 different categories, 437,513 training images and 24,426 test images. Each image has one ground truth label. The iNat dataset is highly imbalanced with dramatically different number of images per category and the imbalance ratio ρ is 500.

C.2 AUGMENTATION

we incorporated ten commonly used DA methods in our experiments, and descriptions are shown in Table 4. The specific code implementation can be found in '/aug/doda.py'.

Table 4: Description of DAs utilized in DODA.

DA	Parameter	Description
Flip	0/1	Flip top and bottom
Mirror	0/1	Flip left and right
EdgeEnhance	0/1	Increasing the contrast of the pixels around the targeted edges
Detail	0/1	Utilize convolutional kernel [[0,-1, 0], [-1, 10,-1], [0,-1, 0]]
Smooth	0/1	Utilize convolutional kernel [[1, 1, 1], [1, 5, 1], [1, 1, 1]]
AutoContrast	0/1	Remove a specific percent of the lightest and darkest pixels
Equalize	0/1	Apply non-linear mapping to make uniform distribution
Invert	0/1	Negate the image
GaussianBlur	[0, 2]	Blurring an image using Gaussian function
Rotate	[0, 30]	Rotate the image

C.3 BASELINES

To ensure a fair comparison, we select a large number of long-tailed learning methods as baselines in our experiments, and integrate DODA with these baselines to evaluate the effectiveness and flexibility of DODA. In addition, we also select the state-of-the-art data augmentation methods as comparison baselines to prove the superiority of DODA in long-tailde learning.

Long-tailed methods:

- CE He et al. (2016): is a cross-entropy loss based model, which is one of the most classic methods in the field of deep long-tailed learning.
- CE-DRW Cao et al. (2019): is a two-stage fine-tuning strategy based on cross-entropy loss.
- LWS Kang et al. (2020): is a two-stage training strategy, which keeps both the representations and classifier weights fixed and only learn the scaling factors.

- cRT Kang et al. (2020): is a two-stage training strategy, which keeps the representations fixed and randomly re-initialize and optimize the classifier weights using class-balanced sampling.
- LDAM-DRW Cao et al. (2019): extends the existing soft margin loss by enforcing class-dependent margins based on label frequencies and further introduces a deferred re-balancing optimization schedule.
- BS Ren et al. (2020): proposes to use the label frequencies to adjust mode predictions during training, so that the bias of class imbalance can be alleviated by the prior knowledge.
- RIDE (3 experts) Wang et al. (2021): introduces a knowledge distillation multi-expert framework to reduce the parameters by learning a student network with fewer experts.
- BCL Zhu et al. (2022): proposes a balanced contrastive learning loss and learns stronger feature representations through a dual-branch framework.
- CMO Park et al. (2022): focuses on utilizing the rich context of majority samples to improve the diversity of minority samples and mixes minority and majority images by using CutMix to enhance balancing and robustness simultaneously.
- CUDA Ahn et al. (2023): is a simple and efficient curriculum, which is designed to find the appropriate per-class strength of data augmentation.

Data augmentation methods:

- AutoAugment Cubuk et al. (2019): describes a simple procedure to automatically search for improved data augmentation policies by designing a search space where a policy consists of many sub-policies, one of which is randomly chosen for each image in each mini-batch.
- Fast AutoAugment Lim et al. (2019): finds effective augmentation policies via a more efficient search strategy based on density matching.
- DADA Li et al. (2020a): relaxes the discrete DA policy selection to a differentiable optimization problem via Gumbel-Softmax and introduces an unbiased gradient estimator to learn an efficient and accurate DA policy.
- RandAugment Cubuk et al. (2020): proposes a simplified search space that vastly reduces the computational expense of automated augmentation, and permits the removal of a separate proxy task.

D MORE EMPIRICAL RESULTS

D.1 PARAMETER SENSITIVITY ANALYSIS OF p_{aug}

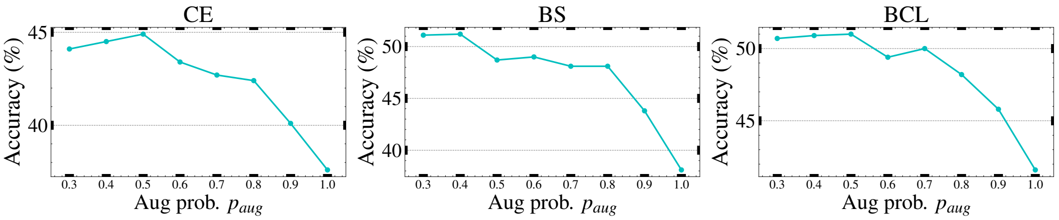

Figure 8: Parameter sensitivity analysis of augmentation probability p_{aug} on CE, BS, and BCL.

To preserve the knowledge of the original dataset, we define the augmentation probability p_{aug}. For further analyze the impact of p_{aug}, we conduct a sensitivity analysis of hyperparameter p_{aug}. As shown in Figure 8, we test 8 different hyperparameter settings on three baselines, and the experimental results showed that a too small augmentation probability cannot sufficiently improve the model's generalization, while a too large augmentation probability cannot retain the knowledge in the original dataset, resulting in a decrease in model performance.

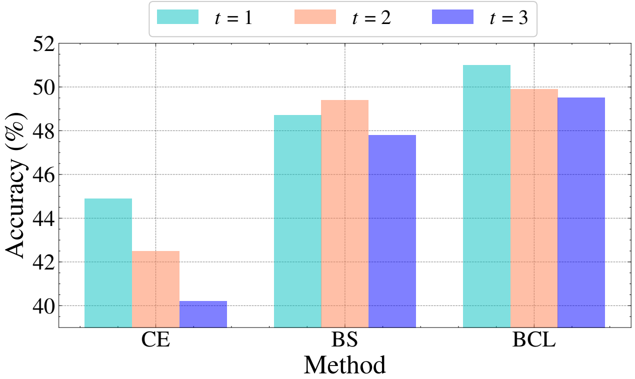

Figure 9: Parameter sensitivity analysis of number of DAs be selected t.

D.2 PARAMETER SENSITIVITY ANALYSIS OF t

In previous analyses, we select the optimal DA for each class. However, we find that in some baselines, multiple DAs can be beneficial. Therefore, we further conduct a parameter sensitivity analysis on the number of DAs be selected t. As shown in Figure 9, we test three hyperparameter settings (t = 1, 2, 3) on three baselines. It can be observed that on CE and BCL, the model tends to select the optimal DA, while on BS, it tends to select both the optimal and suboptimal DAs. This phenomenon is consistent with the trend of selection hierarchies during training mentioned in the main text.

D.3 PARAMETER SENSITIVITY ANALYSIS OF NUM. OF DAS

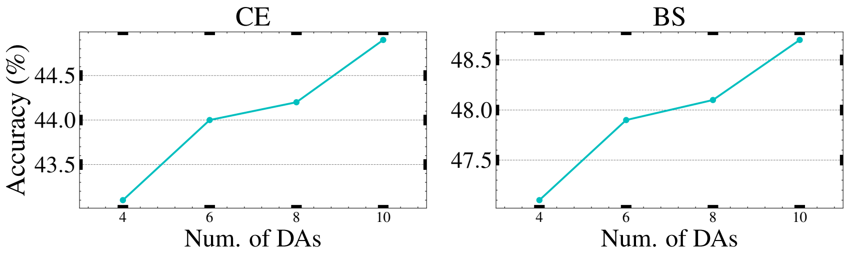

Figure 10: Impact of different numbers of DAs CIFAR-100-LT (IR=100).

As shown in Figure 10, we gradually reduced the number of DAs based on the degree of preference. The results indicate that reducing the number of augmentations leads to a loss of diversity. However, when the 'neglected' augmentations are removed, the model performance does not significantly degrade.

D.4 NETWORK ARCHITECTURE ANALYSIS

As shown in Figure 11, following Ahn et al. (2023), we also utilize ResNet-10 Liu et al. (2019) and ResNeXt-50 Xie et al. (2017) as our backbone network on ImageNet-LT. We conduct comparative experiments on three baselines (e.g., CE, BS, and BCL), and the results show that no matter what kind of backbone is used, DODA can always bring stable improvement to long-tailed learning algorithms.

D.5 TRAINING TIME ANALYSIS

In DODA's augmentation pipeline, we require additional computations to update and maintain the augmentation preference list for each class. Therefore, compared to the original baselines, using

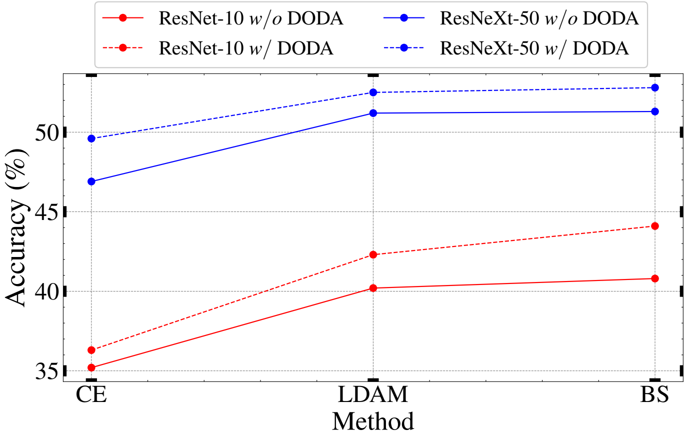

Figure 11: Network architecture analysis.

DODA incurs additional training time. As shown in Table 5, using DODA inevitably brings varying degrees of computational cost, but these costs are acceptable. For example, BS *w/o* DODA achieves better model performance and avoids serious sacrifices with only × 0.09 additional cost.

Table 5: Training time (min.) analysis on various algorithms.

Method	CE	BS	BCL
w/o DODA	60	68	94
w/ DODA	68 (× 1.13)	74 (× 1.09)	102 (× 1.09)

D.6 MORE ANALYSIS ON SACRIFICE RATES

SR on Different Long-tailed Baselines: We provided the sacrifice rates of different data augmentations on CE in Figure 2, indicating that DA can lead to sacrifice problems for the original baseline. Similarly, long-tailed learning baselines also face this issue. Based on your comments, we have conducted further experiments on cRT and CIFAR-100-LT dataset (IR = 100). The results in Table 6 show that CUDA improved accuracy while sacrificing performance for certain classes, while DODA mitigated this sacrifice issue by preserving performance across classes.

Table 6: Accuracy (%) on CIFAR-100-LT dataset (IR = 100) wtih cRT. SR (%) indicates the sacrifice rate.

Method	Head	Medium	Tail	All	SR
CE	65.6	36.2	8.2	38.1	-
CE + CUDA	70.7	41.4	9.3	42.0	29
CE + DODA	74.8	43.8	10.0	44.5	5
cRT Kang et al. (2020)	64.4	49.1	25.8	47.5	-
cRT + CUDA	63.2	50.9	26.6	47.9	22
cRT + DODA	64.4	51.2	27.5	48.7	6

In general, just like focusing on tail classes when improving the average accuracy, when applying DAs in long-tailed learning, focusing on vulnerable classes that are easy to be sacrificed is also in line with the purpose of long-tailed learning.

SR on Different DA Baselines: We also tested different class-independent techniques (e.g., AutoAugment, CutOut) to demonstrate the superiority of our method. The specific experimental results are shown in Table 8.

AutoAugment improves the average accuracy on cRT and takes effect on each shot. However, we further analyze the sacrifice problem caused by DAs, and we find that despite achieving good performance, AutoAugment still cannot avoid the sacrifice problem, which means,

- The performance improvement of AutoAugment is hypocritical, for example, in the tail classes, the model achieves performance gains on some classes, while performing badly on others (i.e., pleasing the 'strong' and bullying the 'weak'). This sacrifice goes against the purpose of long-tailed learning despite the average performance improvement of the model on the tail classes.
- Both class-independent techniques lead to the sacrifice problem of sacrifice. From the sacrifice rate of different shots, it can be found that compared with the head classes, more classes in the tail classes are sacrificed, indicating that the tail classes are more likely to be regarded as the bullied 'weak' mentioned above.

Table 7: Accuracy (%) on CIFAR-100-LT dataset (IR = 100) wtih cRT. $(\cdot)$ indicates the sacrifice rate of different shots.

Method	Head	Medium	Tail	All
cRT Kang et al. (2020)	64.4(-)	49.1(-)	25.8(-)	47.5(-)
cRT + AutoAugment	64.8(5)	49.9(6)	25.9(13)	47.9(24)
cRT + CutOut	61.3(12)	44.5(15)	21.7(23)	43.6(50)
cRT + DODA	64.4(2)	51.2(1)	27.5(3)	48.7(6)

SR on Different Epochs: Here, we analyzed the changes in the sacrifice rate. The results shown in Table 2 show that the sacrifice problem caused by previous DAs cannot be eliminated during training, while DODA significantly improves this.

Table 8: Sacrifice rate (%) on Various Epochs.

Epoch	100	200	400
CE + RandAugment	328	315	309
CE + DODA	73	59	55

D.7 MORE COMPARISONS WITH MODIFIED TWO-STAGE MODEL

Here, we compare DODA with CC-SAM Zhou et al. (2023), which is a two-stage model improvement method that trains the model in a decoupled manner and introduces class-conditional sharpness-aware minimization in the first stage. We have improved the existing open-source implementation and incorporated DODA's augmentation strategy. The quantitative experimental results are shown in Table 9.

Table 9: Accuracy (%) on CIFAR-100-LT dataset (Imbalance ratio = 100) wtih CC-SAM. SR (%) indicates the sacrifice rate.

Method	Head	Medium	Tail	All	SR
CC-SAM Zhou et al. (2023)	67.6	51.2	30.5	50.7 (+ 0.0)	-
CC-SAM + CUDA Ahn et al. (2023)	67.5	52.0	30.7	51.0 (+ 0.3)	31
CC-SAM + DODA	68.4	53.7	33.6	52.8 (+ 3.1)	6

D.8 More Comparisons with Auto DA in other fields

In this section, we compare DODA with Auto DA selection algorithms in other fields. Here we choose the most advanced method Zaiem et al. (2022) in the speech field as a comparison. However, directly applying the complete method from it in long-tailed learning does not lead to fair comparisons. So we partially implemented the augmentation strategies proposed in Zaiem et al. (2022). Firstly, since Zaiem et al. (2022) relies on a carefully designed pretext task, we replaced it with contrastive learning using cropping and augmentation, where pretext labels for each augmented view of a sample corresponding to the ID of the sample it originated from. Then, we replaced the downstream task related to speech with a long-tailed classification task. The experimental results of this modified implementation are shown in Table 10.

It can be observed that using the automatic augmentation strategy from Zaiem et al. (2022) results in limited performance improvement, while our method outperforms it significantly. The reasons for this are as follows: (1) Zaiem et al. (2022) relies on a carefully designed pretext task, so the improvement it brings may come from diversified data augmentation. (2) Zaiem et al. (2022) lacks the necessary focus on the tail classes, while our method pays more attention to inter-class fairness, resulting in better performance.

Table 10: Accuracy (%) on CIFAR-100-LT dataset (Imbalance ratio = 100) wtih Zaiem et al. (2022).

Method	Head	Medium	Tail	All
CE	65.6	36.2	8.2	38.1 (+ 0.0)
CE + Zaiem et al. (2022)	68.9	38.7	8.4	40.2 (+ 2.1)
CE + DODA	74.8	43.8	10.0	44.5 (+ 6.4)

D.9 Exploration of Combinations with SOTA Long-tailed DA

From the macro perspective of long-tailed learning, both DODA and CUDA belong to dynamic DA. However, at the methodology level, the two are different. A simple comparison is as Table 11:

Table 11: Comparison at the methodology level.

Method	Adaptive Strength	Adaptive Function	Inter-class Fairness	Cold-boot Issues
CUDA	✓			
DODA		✓	✓	✓

It can be seen that to ensure fairness between classes while improving accuracy, we have made some methodology-level improvements. More interestingly, we find that CUDA and DODA are orthogonal, and we can find the optimal DA function and strength at the same time. The exploratory results are as follows:

Table 12: Exploratory results (%) on CIFAR-100-LT dataset (IR = 100).

Method	Head	Medium	Tail	All
CE + DODA	74.8	43.8	10.0	44.5
CE + DODA + CUDA	74.7	44.1	10.2	44.6

Although the performance gain is limited, continuing to explore this compositionality is beneficial for long-tail learning.

D.10 MORE TRENDS OF THE SELECTION HIERARCHIES ON DIFFERENT INDEXS

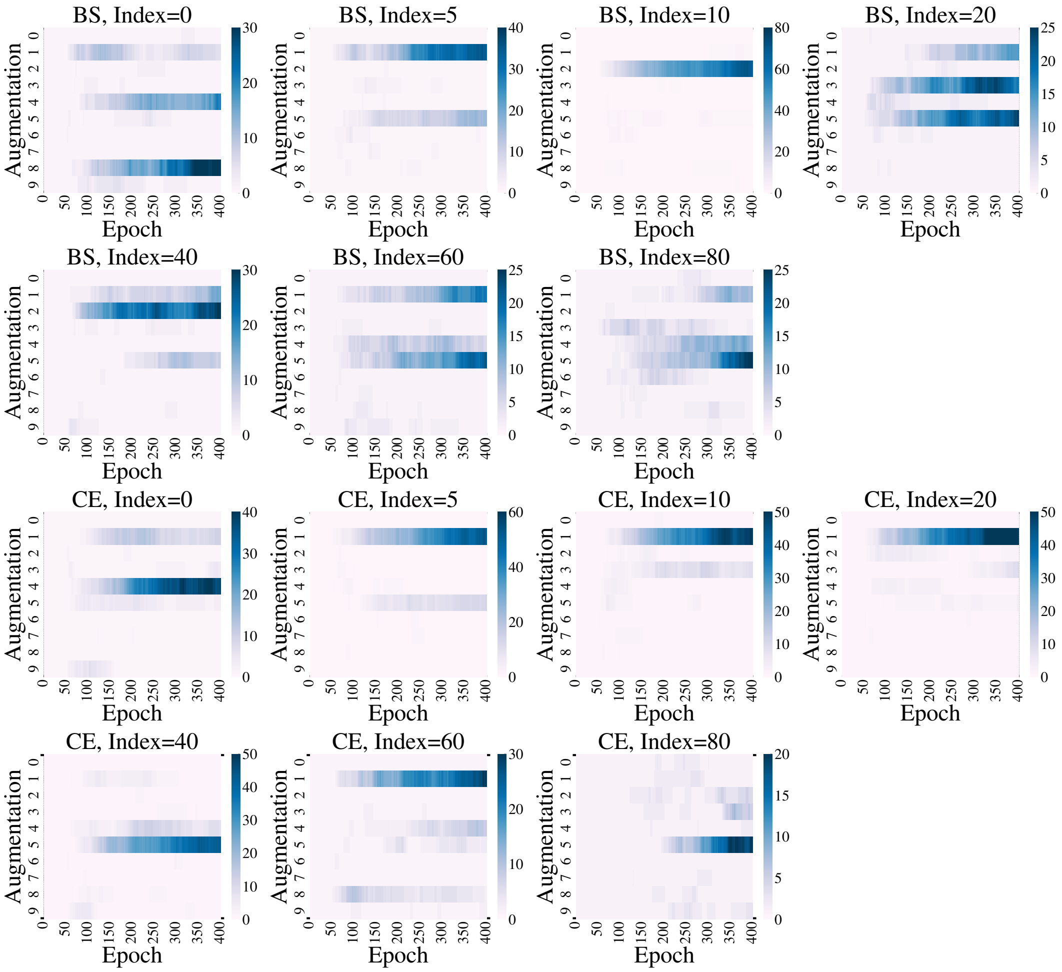

Figure 12: Trends of the selection hierarchies on different indexs.