Class-Balanced Loss Based on Class Volume for Long-Tailed Object Recognition

ZhiJian Zheng
zhijian@comp.nus.edu.sg
Teck Khim Ng
ngtk@comp.nus.edu.sg
Department of Computer Science
National University of Singapore

Abstract

The performance of classification neural networks is often suboptimal in the real world due to long-tailed data distributions. Re-sampling and re-weighting based on class frequency have been adopted in the literature to address the long-tailed problem. In this paper, we focus on the re-weighting approach. Re-weighting factors estimated by state-of-the-art approaches are determined by the number of samples which ignore the within class diversity (e.g. the cat class is visually more diverse than the frog class). In this paper, we propose a concept called class volume that measures the within class diversity and use this class volume to dynamically adjust the per-class weight. Our method does not introduce any hyperparameter and can be easily integrated into existing models with little computation overhead. We conducted extensive experiments and set the new state-of-the-art performance on widely-used long-tailed recognition benchmarks.

1 Introduction

Research on neural networks mostly focus on learning from balanced datasets where each class has approximately the same number of samples. Real-world data however is often long-tailed by nature where the number of samples per class varies significantly. A naively learned model tends to be biased towards the head classes resulting in poor performance for the tail classes.

In general, there are two strategies to alleviate the challenge of long-tailed data problems: re-balancing data distribution; and transfer learning. Re-balancing methods work either by re-sampling (i.e., under-sampling or over-sampling) the data to achieve equal class frequency or re-weighting the loss function in training. Transfer learning methods attempt to transfer knowledge from the head classes to the tail classes.

Under-sampling risks missing the important concepts while over-sampling risks overfitting. Transfer learning methods require the design of task-specific network models which are usually hard to generalize. In this paper, we focus on the re-weighting approach.

The re-weighting approach aims to adjust the loss of each sample with a different weight to shift the decision boundary. The inverse class frequency is first adopted in [16, 37]. Later on, Cui *et al*. proposed the concept of effective number of samples [8] which is defined as the hypothetical volume covered by the samples of each class. The effective number takes

into consideration the diminishing benefit as the number of samples increases. We view the effective number as a coarse approximation of our class volume concept that we introduce in this paper.

We conceptualize the class volume as a continuous volume defined by the distribution of the training data from a class. We propose to estimate the class volume as the product of standard deviations on each axis in the logits feature space.

We view the class volume as mainly determined by two factors:

1. The within class diversity.
2. The number of samples used to train the classifier.

The effective number approach [8], along with other class frequency based re-weighting approaches [16, 26, 28, 37], only modeled the second factor but ignored the within class diversity.

We conducted extensive experiments on the long-tailed CIFAR (CIFAR-LT) [8] datasets to study the various design choices as well as to compare with existing re-weighting techniques. We finalized our design choice from the experiment results on the CIFAR-LT datasets and then integrated our method with the state-of-the-art transfer learning model VL-LTR [35]. We then conducted extensive experiments on three commonly used large-scale datasets: ImageNet-LT [23], Places-LT [23], and iNaturalist 2018 [14].

Our method has three advantages: 1) the classification accuracy is improved significantly over existing re-weighting methods; 2) our method is adaptive to the within class distribution without the need to fine-tune any hyperparameter; 3) our method can be easily integrated into existing models with little overhead. We elevated the state-of-the-art performance by integrating our method into one of the best-performing visual-linguistic transfer learning models VL-LTR [35].

2 Related Works

Re-sampling One direct approach to deal with an imbalanced dataset is to re-sample the data. Resampling can be done by either sub-sampling to remove samples from the head classes or over-sampling to add repeated samples from the tail classes. However, sub-sampling risks missing the important concepts while over-sampling risks overfitting. Estabrooks *et al*. [9] studied the two resampling methods with different resampling rates and demonstrated that the overall classification performance can be improved with a proper combination of different expressions of the resampling approaches.

Novel samples can be synthesized [4, 10, 32] to mitigate the overfitting problem. The noise introduced however also hinders the performance.

Re-weighting Another approach is cost-sensitive learning which re-weights the loss function. The inverse class frequency is used in [16, 37]. The inverse square root of class frequency is used in [26, 28]. The re-weighting factors in these works are empirically chosen. Later on, Cui *et al*. proposed the concept of effective number [8] of samples which was defined as the hypothetical volume covered by the samples of each class. Sound theoretical analysis was provided for the effective number definition. Jamal *et al*. interpreted the class balanced loss function from the domain adaptation perspective [18]. Label-distribution-aware margin (LDAM) [2] gives the tail classes larger margins. Distribution alignment (DisAlign) [40] adjusts the classification scores to align the model prediction with a weighted

empirical distribution on the training set. LTR-Weight-Balancing [1] balances norms of per-class network weights by parameter regularization.

Finer-grained weights can be assigned on a per-sample basis. Focal loss [23] re-weights each sample by inverse prediction probability. Jamal *et al.* [18] learns the weight for each sample from the data. Influence-balanced (IB) loss [29] down-weights the head class samples around the decision boundary to create a smoother decision boundary.

The re-weighting factors can also be learned [18, 33] from a balanced meta validation set. The meta validation set however also increases the burden on the tail classes as the number of samples in the tail classes is already very limited.

There are also studies targeting non-balanced testing data distribution. Label distribution disentangling (LADE) loss [13] disentangles the model prediction from the training data distribution. The learned model can then be calibrated for arbitrary test data distribution. Test-time aggregating diverse experts (TADE) [41] uses multiple network branches to handle agnostic test data distribution.

Transfer learning Transfer learning techniques, such as head-to-tail knowledge transfer [3, 7, 15, 37, 38] or knowledge distillation [12, 17, 21, 36], have been explored to address the long-tail problem. Recent visual-linguistic models [24, 25, 35] achieve the state-of-the-art by utilizing the text modality. The text descriptions of a class are used to guide the network's attention to those visual features that relate to the class. This is especially useful for large-scale high-resolution datasets which are rich in visual features. These methods however require the design of task-specific models which are usually hard to be generalized to different tasks.

Although the latest transfer learning methods have outperformed the re-weighting methods, we argue that the re-weighting approaches are still useful in the long-tail recognition field due to their generalization capacity which makes them simple to be integrated into existing models (including transfer learning methods). In this paper, we focus on the re-weighting technique. Our method does not add significant implementation effort or computation overhead. In our experiments, we demonstrated that our method outperforms other re-weighting methods. We also set the new state-of-the-art performance by integrating our method into the latest visual-linguistic model [35].

3 Method

Let n_i be the number of training samples in the ith class. $i \in \{1,2,...,C\}$ where C is the total number of classes. Let D be the dimension of the feature space.

We define the $D \times n_i$ feature matrix of the ith class as:

$$\vec{F}_i = [\vec{f}_{i,1}, \vec{f}_{i,2}, \ldots, \vec{f}_{i,j}, \ldots, \vec{f}_{i,n_i}] \tag{1}$$

where $\vec{f}_{i,j} \in \mathbb{R}^D$ is a column vector representing the features of a training sample j in class i.

The mean of the feature vectors from the ith class can be estimated as:

$$\vec{\mu}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} \vec{f}_{i,j} \tag{2}$$

The standard deviation of the feature vectors for the ith class can be estimated as:

$$\vec{\sigma}_i = \sqrt{\frac{1}{n_i - 1} \sum_{j=1}^{n_i} (\vec{f}_{i,j} - \vec{\mu}_i)^2} \tag{3}$$

We define a new concept called class volume as the product of the standard deviations of a particular class (Equation 4).

$$v_i = \prod_{k=1}^{D} \sigma_{i,k} \tag{4}$$

To balance the loss, let α_i be the weighting factor for the ith class. α_i can be considered as a training sample's power of influence to push the decision boundary away from the class center. The aggregated power of influence $(n_i\alpha_i)$ for each class measures the distance from the decision boundary to the class center. It is thus desired to set $(n_i\alpha_i)$ to match the radius of the class volume, i.e.

$$n_i\alpha_i \propto \sqrt[D]{v_i} \tag{5}$$

The weighting factor can be calculated as:

$$\alpha_i = \beta \frac{\sqrt[D]{v_i}}{n_i} \tag{6}$$

where β is a normalization factor (Section 3.2).

3.1 Choices of Feature Space

We have designed three variants of feature space to calculate the class volume.

Distribution Variance (DV): It is natural to choose the logits as the feature vector since the loss function is directly applied on the logits. In such a case, the dimension of the feature space is the same as the number of classes, i.e. $D = C$.

Principal axis Variance (PV): Principal Component Analysis (PCA) can be applied to analyze correlations between axes. A tighter version of the class volume can be estimated as the product of the standard deviation of the feature vectors projected on the principal component axes. The projected feature matrix is:

$$\vec{F}_i^{\,\dagger} = \vec{W}_i^{\,\dagger} \vec{F}_i \tag{7}$$

where $\vec{W}_i^{\,\dagger}$ is a $D^\dagger \times D$ matrix whose rows are the first $D^\dagger$ principal component axes with the biggest variance.

Decision Boundary axis Variance (BV): Classification is neither done in the Euclidean space nor the PCA space. The class label is chosen as the one with the highest logit. The decision boundary between class i and class j is a $C-1$ dimensional hyperplane defined by the axis $\vec{e}_i - \vec{e}_j$ where $\vec{e}_i$ denotes the C-dimensional one-hot vector with 1 at position i. For each class, there are $C-1$ hyperplanes that separate it from other classes. We can project

the feature vectors onto the $C-1$ axes defined by those separation planes. The projection is defined as a $(C-1)\times C$ transformation matrix $\vec{W}_i^{\ddagger}$:

$$\vec{W}_i^{\ddagger} = \begin{bmatrix} \vec{e}_i - \vec{e}_1 \\ \cdots \\ \vec{e}_i - \vec{e}_j \\ \cdots \\ \vec{e}_i - \vec{e}_C \end{bmatrix} \tag{8}$$

where $j \neq i$. The projected feature matrix is:

$$\vec{F}_i^{\ddagger} = \vec{W}_i^{\ddagger} \vec{F}_i \tag{9}$$

3.2 Weights Normalization Factor

Previous works [8, 18] normalize the weights to keep the total loss at the same scale after re-weighting. We agree with this principle as it helps to ensure that the hyperparameters (such as learning rate, weight decay, etc) remain optimal. The previous works [8, 18] achieve this principle by making the mean of the normalized weights equal to 1. We noticed however that such normalization is non-optimal when the data is highly skewed.

We instead formulate the *same scale loss* principle as follows:

$$E(\alpha\ell) = E(\ell) \tag{10}$$

where α and ℓ are the weighting factor and loss for one training sample respectively. $E()$ denotes the expected value.

Assuming ℓ and α are independent random variables. Equation 10 reduces to:

$$E(\alpha) = 1 \tag{11}$$

Since α is a discrete random variable, the probability density $P(\alpha_i)$ can be calculated by counting. Therefore:

$$E(\alpha) = 1 \tag{12}$$

$$\implies \sum_{i=1}^{C} P(\alpha_i)\alpha_i = 1 \tag{13}$$

$$\implies \sum_{i=1}^{C} \frac{n_i}{\sum_{j=1}^{C} n_j}\alpha_i = 1 \tag{14}$$

$$\implies \sum_{i=1}^{C} n_i\alpha_i = \sum_{j=1}^{C} n_j \tag{15}$$

The normalization factor β can then be derived by substituting Equation 6 into Equation 15:

$$\beta = \frac{\sum_{j=1}^{C} n_j}{\sum_{i=1}^{C} \sqrt[D]{v_i}} \tag{16}$$

The normalization factor β is used in Equation 6 to compute the normalized per-class weights.

Dataset	CIFAR-LT-10					CIFAR-LT-100				
Imbalance	200	100	50	20	10	200	100	50	20	10
CE	39.9 ± 2.1	26.5 ± 0.5	20.5 ± 0.3	16.2 ± 0.4	12.9 ± 0.3	63.0 ± 0.4	57.8 ± 0.3	53.0 ± 0.3	46.5 ± 0.3	40.9 ± 0.3
EN [8]	37.7 ± 2.1	27.3 ± 2.6	19.1 ± 0.3	14.8 ± 0.4	12.4 ± 0.3	61.7 ± 0.5	56.9 ± 0.4	52.1 ± 0.4	44.4 ± 0.4	40.5 ± 0.3
DV (Ours)	27.3 ± 1.4	**20.8** ± 0.3	**16.5** ± 0.3	13.9 ± 0.4	11.8 ± 0.2	**56.9** ± 0.4	**52.9** ± 0.4	**47.8** ± 0.4	**42.7** ± 0.3	38.6 ± 0.3
DV best (Ours)	**24.9**	**20.2**	**16.1**	**13.2**	**11.2**	**56.4**	**51.8**	**47.2**	**41.8**	**37.9**
Focal loss [23] *	34.7	29.6	23.3	17.2	13.3	64.4	61.6	55.7	48.0	44.2
Meta-Weight-Net [31] *	32.8	26.4	20.9	15.6	12.5	63.4	58.4	54.3	47.0	41.1
DA with CE [18] *	29.3	23.6	19.5	**13.5**	**11.2**	60.7	56.7	51.5	44.4	40.4
IB [24] *	**26.0**	21.7	18.3	14.2	11.8	62.7	57.9	53.8	47.4	42.9
LDAM [1] *	-	26.7	-	-	13.0	-	60.4	-	-	43.1
LDAM-DRW [1] *	-	23.0	-	-	11.9	-	58.0	-	-	41.3
LADE [11] *	-	-	-	-	-	-	54.6	49.5	-	**38.3**

Table 1: Classification error of ResNet-32 on CIFAR-LT datasets [8]. * indicates results reported in original paper. "CE" means the cross-entropy training.

3.3 Stability of the Estimated Variance

Statistical estimation only gets stable when there are sufficient samples. In some datasets, our estimated variance might be noisy for the last few tail classes. In such a case, we may choose to discard those noisy estimations and substitute them with the average estimations from the other classes.

We observed that the rare classes with few samples are more likely to have a larger class volume. Intuitively, this is because the training process spends less effort on rare classes. Therefore, the resulting representation is more sparse. Nevertheless, a tail class will be assigned a larger weight even if we approximate its class volume using an average value. This is because the weight is inversely proportional to the number of samples in the class (Equation 6).

4 Experiments

We perform extensive experiments on the artificially created CIFAR-LT datasets [8] and three commonly used large-scale datasets: ImageNet-LT [23], Places-LT [23], and iNaturalist 2018 [14]. Following [8], the imbalance factor (IF) of a dataset is defined as the ratio of the class size between the most frequent class and the least frequent class. i.e., $IF = \frac{\max(n_i)}{\min(n_i)}$ where n_i is the number of training samples in the ith class.

4.1 Experiments on CIFAR-LT

The original CIFAR [19] dataset contains 50,000 training images and 10,000 test images. The dataset is balanced in the sense that each class has the same number of samples. Following common practice [8, 18], long-tailed CIFAR datasets are created by sub-sampling the original dataset according to the exponential distribution. Five training sets are created with the imbalance factors ranging from 10 to 200. The test set is balanced and remains unchanged.

We run both comparison experiments and ablation studies with the CIFAR-LT datasets. We implemented our method on top of Jamal *et al.*'s code [18]. We followed the same training hyperparameter settings. Specifically: 32 layers residual network (ResNet-32) [10] is used as the backbone; the network is trained for 200 epochs; the learning rate is initialized as 0.1 and decayed by 0.01 at the 160th and 180th epochs; the batch size is 100.

Dataset	CIFAR-LT-10					CIFAR-LT-100				
Imbalance	200	100	50	20	10	200	100	50	20	10
CE	39.9±2.1	26.5±0.5	20.5±0.3	16.2±0.4	12.9±0.3	63.0±0.4	57.8±0.3	53.0±0.3	46.5±0.3	40.9±0.3
DV	30.4±1.7	21.3±0.4	16.8±0.3	14.4±0.5	11.9±0.3	60.5±0.4	54.1±0.5	48.7±0.4	43.2±0.4	38.9±0.4
PV	30.5±1.7	21.3±0.5	16.9±0.3	14.3±0.4	11.9±0.3	62.3±0.4	57.1±0.4	51.5±0.4	46.6±0.4	43.4±0.4
PV ($D^{\dagger}=10$)	-	-	-	-	-	60.0±0.5	54.9±0.3	48.3±0.4	42.7±0.4	38.5±0.4
BV	30.8±1.9	21.2±0.4	16.9±0.3	14.4±0.4	11.8±0.3	60.1±0.3	54.2±0.4	48.7±0.4	43.3±0.4	38.9±0.3

Table 2: Ablation study of feature space choices on CIFAR-LT datasets. The results are test top-1 errors%. "CE" means the cross-entropy training.

Dataset	CIFAR-LT-10					CIFAR-LT-100				
Imbalance	200	100	50	20	10	200	100	50	20	10
CE	39.9±2.1	26.5±0.5	20.5±0.3	16.2±0.4	12.9±0.3	63.0±0.4	57.8±0.3	53.0±0.3	46.5±0.3	40.9±0.3
DV	30.4±1.7	21.3±0.4	16.8±0.3	14.4±0.5	11.9±0.3	60.5±0.4	54.1±0.5	48.7±0.4	43.2±0.4	38.9±0.4
DV +NE	27.3±1.4	20.8±0.3	16.5±0.3	13.9±0.4	11.8±0.2	56.9±0.4	52.9±0.4	47.8±0.4	42.7±0.3	38.6±0.3
DV +NE +S30	-	-	-	-	-	56.7±0.4	52.6±0.4	47.8±0.4	42.8±0.3	-

Table 3: Ablation study of contribution factors on CIFAR-LT datasets. The results are test top-1 errors%. "CE" means the cross-entropy training.

Table 1 shows the classification errors of ResNet-32 on the CIFAR-LT datasets with different imbalance factors. We run each experiment 30 times and report the mean, standard deviation, and the best of the classification errors. Our method outperforms other re-weighting methods by a large margin. Our advantage is even more pronounced on the more challenging CIFAR-LT-100 dataset when the data is highly skewed (50 - 200 imbalance factor).

Ablation study: choices of feature space Table 2 shows the classification errors of ResNet-32 on the CIFAR-LT datasets with different feature space design choices as explained in Section 3.1. The principal axis variance (PV) variant performs poorer on the CIFAR-LT-100 dataset which has a smaller number of samples (3 to 500) per class. The relatively smaller number of samples per class leads to an unstable estimation in the last few principle axes. The problem can be rectified by restricting the principal axes to the first 10 principle axes with the biggest variance. After the rectification, the PV ($D^{\dagger}=10$) variant gains a slight advantage over the other variants.

The re-weighting methods essentially attempt to adjust the decision boundary according to the class boundaries. These re-weighting methods mainly differ in the way how the class boundary is estimated. We use the class volume concept defined by the within-class variance to estimate the class boundary. Principal axis variance gives a better estimation if features in different dimensions are correlated. Decision boundary axes variance treats variations on non-decision boundary axes as noise. Our experiment results indicate no significant difference among the three variants. That indicates that the features in different dimensions have low correlations. Our original choice of the simplest distribution variance (DV) suffices. We use the DV variant in all the subsequent experiments.

Ablation study: contribution factors Our method consists of three components: DV estimates the class volume by distribution variance; NE normalizes the weight so that the scale of the expected loss remains unchanged; S30 substitute the class volume with the average of that of other classes when a class is under-represented (number of samples is less than 30). Table 3 examines the components by applying them one after another. DV and NE contribute significantly to the classification accuracy especially when the dataset is highly skewed. NE's contribution increases as the imbalance factor increases. That is because it

	iNaturalist 2018		ImageNet-LT		Places-LT	
Method	Backbone	Acc(%)	Backbone	Acc(%)	Backbone	Acc(%)
LADE [13] *	R-50	70.0	X-50	53.0	R-152	38.8
LTR-Weight-Balancing [1] *	R-50	70.2	X-50	53.9		
SSD [21] †	R-50	71.5	R-50	56.0		
MiSLAS [42] *	R-50	71.6	R-50	52.7	R-152	40.4
LACE [27] *	R-152	72.0	R-152	52.1		
ResLT [6] †	R-50	72.3	X-101	55.1	R-152	39.8
CMO + RIDE [30] *	R-50	72.8	R-50	56.2		
TADE [41] †	R-50	72.9	X-50	58.8	R-152	40.9
RIDE (4 experts) [36] †	R-50	73.2	R-50	55.4		
PaCo [5] †	R-50	73.2	X-101	60.0	R-152	41.2
DiVE + RIDE [17] *	R-50	73.4	X-50	57.1		
DisAlign [40] *	R-152	74.1	X-50	53.4	R-152	39.3
BatchFormer + RIDE [15] *	R-50	74.1	R-50	55.7		
BatchFormer + PaCo [15] *			R-50	57.4	R-152	41.6
NCL + Ensemble [20] *	R-50	74.9	R-50	59.5	R-152	41.8
CBD + Ensemble [17] *	R-101	75.3	R-152	57.7		
BALLAD [25] *			RN50×16	76.5	ViT-B	49.5
VL-LTR [35]	R-50	74.4	R-50	69.9	R-50	48.1
DV + VL-LTR (ours)	R-50	**76.4**	R-50	**70.3**	R-50	**48.7**
VL-LTR [35]	ViT-B	76.0	ViT-B	77.0	ViT-B	50.2
DV + VL-LTR (ours)	ViT-B	**78.0**	ViT-B	**77.4**	ViT-B	**50.8**
RAC [24] *	ViT-B	**80.2**			ViT-B	47.2

Table 4: Results on iNaturalist 2018, ImageNet-LT, and Places-LT. "R-*" means the ResNet [11] backbone. "X-*" means the ResNeXt [39] backbone. "RN50×16" [31] is 16× computation cost of ResNet-50 following the style of EfficientNet [34]. "*" indicates results copied from the original paper. "†" indicates results copied from [35].

helps to keep the total loss roughly at the same scale when the data is highly skewed. The contribution of the S30 step is relatively small because only a small portion of the training samples is affected by the S30 step.

4.2 Experiments on iNaturalist 2018, ImageNet-LT and Places-LT

We integrate our method into one of the best-performing visual-linguistic models VL-LTR [35]. The VL-LTR model has two stages: 1) the pre-train stage learns the visual-linguistic representation; 2) the language guided recognition stage uses the linguistic representations to guide the model's attention to visual features related to each class. We start with the pre-trained stage parameters released by the authors and integrate our method into the language guided recognition stage. We follow the same training hyperparameter settings as the original paper except reducing the number of epochs to 100 for the iNaturalist 2018 dataset. For the ImageNet-LT and Places-LT dataset, all hyperparameters are the same as the original paper including the number of epochs which is 50.

Overall performance Table 4 shows the classification accuracy on the three datasets. For the iNaturalist 2018 dataset, the replicated results (without integrating our method) are slightly lower (74.4 vs 74.6; 76.0 vs 76.8) than that reported in the original paper because

Dataset	Backbone	Method	Overall	Many	Medium	Few
ImageNet-LT	R50	VL-LTR	69.9	78.6	66.3	47.8
		Ours	70.3 ↑	74.4 ↓	70.1 ↑	55.2 ↑
	ViT-B	VL-LTR	77.0	84.8	73.8	57.3
		Ours	77.4 ↑	82.2 ↓	76.6 ↑	64.9 ↑
Places-LT	R50	VL-LTR	48.1	53.0	47.1	36.5
		Ours	48.7 ↑	45.5 ↓	50.3 ↑	48.1 ↑
	ViT-B	VL-LTR	50.2	54.2	48.4	43.4
		Ours	50.8 ↑	49.3 ↓	52.8 ↑	50.0 ↑
iNaturalist 2018	R50	VL-LTR	74.4	78.5	75.1	72.6
		Ours	76.4 ↑	75.2 ↓	76.5 ↑	76.4 ↑
	ViT-B	VL-LTR	76.0	81.1	77.4	73.4
		Ours	78.0 ↑	79.5 ↓	78.3 ↑	77.1 ↑

Table 5: Detailed Per Class Group Accuracy

Dataset	Backbone	Training Time		
		VL-LTR	Ours	Overhead
ImageNet-LT	R50	1:36:13	1:32:10	-4.2%
	ViT-B	2:17:24	2:17:56	+0.4%
Places-LT	R50	0:34:30	0:32:49	-4.9%
	ViT-B	1:00:42	1:01:07	+0.7%
iNaturalist 2018	R50	45:10:40	46:20:47	+2.6%
	ViT-B	43:48:15	45:26:26	+3.7%

Table 6: Training Time

we used shorter training epochs. Our method however can boost the model by a large margin (76.4 vs 74.6; 78.0 vs 76.8) despite having a shorter training epochs (100 vs 360). For the ImageNet-LT and Places-LT datasets, our method boosts state-of-the-art by 0.4 and 0.6 points respectively.

Our method sets the new state-of-the-art performance on the ImageNet-LT and Places-LT datasets. The recent RAC [24] model outperforms the VL-LTR model on the iNaturalist 2018 dataset. We hope to experiment on the RAC model when their code gets released.

Classification bias Following common practices [5, 23, 35], we split each dataset into three subsets by the number of training samples in each class: many-shot (≥ 100 samples), medium-shot ($20 \sim 100$ samples), and few-shot (≤ 20 samples). Table 5 shows the detailed classification accuracy. In all six settings, the performance on classes with "Few" or "Medium" samples was improved, while the performance on head classes with "Many" samples was decreased. That shows that our method indeed reduced the bias. We wish to highlight that the overall performance improved for all datasets as shown in Table 5.

Computational cost The class volume is computed on the entire training set once per epoch. The dimensionality of the logits feature space is not large (less than a few thousands). The computation of variance is also lightweight ($O(N)$). The computational cost for each training sample is therefore a few KFLOPs which is negligible compared to the usual GFLOPs for network training. We extracted the training time from our experiment log and presented it in Table 6. It shows that the computation overhead is insignificant.

5 Conclusion

In this paper, we present a novel concept called class volume as a re-weighting technique to address the long-tailed distribution problem. We define the class volume by feature space distribution variance which captures the bias introduced by imbalance as well as within class diversity. We also propose to normalize the weight base on the expected value. Extensive ablation studies were conducted on the CIFAR-LT dataset to verify the effectiveness of our two contributions in addressing the long-tailed problem.

In addition, our method is adaptive to within class distribution without introducing any additional hyperparameter. Therefore we avoid the tedious hyperparameter fine-tuning process. Our method is lightweight. It can be easily integrated into existing models with little computation overhead. We set the new state-of-the-art performance on widely-used long-tailed recognition benchmarks by integrating our method into the latest visual-linguistic model.

References

[1] Shaden Alshammari, Yu-Xiong Wang, Deva Ramanan, and Shu Kong. Long-Tailed Recognition via Weight Balancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6897–6907, 2022.

[2] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in Neural Information Processing Systems*, 32, 2019.

[3] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing Web-Scale Image-Text Pre-Training To Recognize Long-Tail Visual Concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3558–3568, 2021.

[4] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16(1):321–357, 2002.

[5] Jiequan Cui, Zhisheng Zhong, Shu Liu, Bei Yu, and Jiaya Jia. Parametric Contrastive Learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 715–724, 2021.

[6] Jiequan Cui, Shu Liu, Zhuotao Tian, Zhisheng Zhong, and Jiaya Jia. ResLT: Residual Learning for Long-tailed Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[7] Yin Cui, Yang Song, Chen Sun, Andrew Howard, and Serge Belongie. Large Scale Fine-Grained Categorization and Domain-Specific Transfer Learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4109–4118, 2018.

[8] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-Balanced Loss Based on Effective Number of Samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9268–9277, 2019.

[9] Andrew Estabrooks, Taeho Jo, and Nathalie Japkowicz. A Multiple Resampling Method for Learning from Imbalanced Data Sets. *Computational Intelligence*, 20(1): 18–36, 2004.

[10] Haibo He, Yang Bai, Edwardo A. Garcia, and Shutao Li. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pages 1322–1328. IEEE, 2008.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[12] Yin Yin He, Jianxin Wu, and Xiu Shen Wei. Distilling Virtual Examples for Long-tailed Recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 235–244, 2021.

[13] Youngkyu Hong, Seungju Han, Kwanghee Choi, Seokjun Seo, Beomsu Kim, and Buru Chang. Disentangling Label Distribution for Long-tailed Visual Recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6626–6636, 2021.

[14] Grant Van Horn, Oisin Mac, Alex Shepard, Hartwig Adam, Yang Song, Yin Cui, Chen Sun, Pietro Perona, and Serge Belongie. The iNaturalist Species Classification and Detection Dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8769–8778, 2018.

[15] Zhi Hou, Baosheng Yu, and Dacheng Tao. BatchFormer: Learning to Explore Sample Relationships for Robust Representation Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7256–7266, 2022.

[16] Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. Learning Deep Representation for Imbalanced Classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5375–5384, 2016.

[17] Ahmet Iscen, André Araujo, Boqing Gong, and Cordelia Schmid. Class-Balanced Distillation for Long-Tailed Visual Recognition. In *BMVC*, 2021.

[18] Muhammad Abdullah Jamal, Matthew Brown, Ming-Hsuan Yang, Liqiang Wang, and Boqing Gong. Rethinking Class-Balanced Methods for Long-Tailed Visual Recognition From a Domain Adaptation Perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7610–7619, 2020.

[19] Alex Krizhevsky and Geoffrey Hinton. Learning Multiple Layers of Features from Tiny Images. Technical report, University of Toronto, 2009.

[20] Jun Li, Zichang Tan, Jun Wan, Zhen Lei, and Guodong Guo. Nested Collaborative Learning for Long-Tailed Visual Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6949–6958, 2022.

[21] Tianhao Li, Limin Wang, and Gangshan Wu. Self Supervision to Distillation for Long-Tailed Visual Recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 630–639, 2021.

[22] Tsung Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal Loss for Dense Object Detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

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

[24] Alexander Long, Wei Yin, Thalaiyasingam Ajanthan, Vu Nguyen, Pulak Purkait, Ravi Garg, Alan Blair, Chunhua Shen, and Anton van den Hengel. Retrieval Augmented Classification for Long-Tail Visual Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6959–6969, 2022.

[25] Teli Ma, Shijie Geng, Mengmeng Wang, Jing Shao, Jiasen Lu, Hongsheng Li, Peng Gao, and Yu Qiao. A Simple Long-Tailed Recognition Baseline via Vision-Language Model. *arXiv preprint arXiv:2111.14745*, 2021.

[26] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the Limits of Weakly Supervised Pretraining. In *Proceedings of the European conference on computer vision (ECCV)*, pages 181–196, 2018.

[27] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. In *International Conference on Learning Representations*, 2021.

[28] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, 2013.

[29] Seulki Park, Jongin Lim, Younghan Jeon, and Jin Young Choi. Influence-Balanced Loss for Imbalanced Visual Classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 735–744, 2021.

[30] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6887–6896, 2022.

[31] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *International Conference on Machine Learning*, pages 8748–8763, 2021.

[32] Shiven Sharma, Colin Bellinger, Bartosz Krawczyk, Osmar Zaiane, and Nathalie Japkowicz. Synthetic Oversampling with the Majority Class: A New Perspective on Handling Extreme Imbalance. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 447–456. IEEE, 2018.

[33] Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-weight-net: Learning an explicit mapping for sample weighting. In *Advances in Neural Information Processing Systems*, volume 32, pages 1917–1928, 2019.

[34] Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 6105–6114, 2019.

[35] Changyao Tian, Wenhai Wang, Xizhou Zhu, Xiaogang Wang, Jifeng Dai, and Yu Qiao. VL-LTR: Learning Class-wise Visual-Linguistic Representation for Long-Tailed Visual Recognition. *arXiv preprint arXiv:2111.13579*, 2021.

[36] Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella X. Yu. Long-tailed Recognition by Routing Diverse Distribution-Aware Experts. In *International Conference on Learning Representations*, 2021.

[37] Yu Xiong Wang, Deva Ramanan, and Martial Hebert. Learning to Model the Tail. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[38] Chen Wei, Kihyuk Sohn, Clayton Mellina, Alan Yuille, and Fan Yang. CReST: A Class-Rebalancing Self-Training Framework for Imbalanced Semi-Supervised Learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10857–10866, 2021.

[39] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated Residual Transformations for Deep Neural Networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5987–5995, 2017.

[40] Songyang Zhang, Zeming Li, Shipeng Yan, Xuming He, and Jian Sun. Distribution Alignment: A Unified Framework for Long-tail Visual Recognition. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2361–2370, 2021.

[41] Yifan Zhang, Bryan Hooi, Lanqing Hong, and Jiashi Feng. Test-Agnostic Long-Tailed Recognition by Test-Time Aggregating Diverse Experts with Self-Supervision. *arXiv preprint arXiv:2107.09249*, 2021.

[42] Zhisheng Zhong, Jiequan Cui, Shu Liu, and Jiaya Jia. Improving calibration for long-tailed recognition. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16484–16493, 2021.