Balanced Contrastive Learning for Long-Tailed Visual Recognition

Jianggang Zhu\textsuperscript{1,2,*}; Zheng Wang\textsuperscript{1,2}; Jingjing Chen\textsuperscript{1,2†}; Yi-Ping Phoebe Chen\textsuperscript{3} and Yu-Gang Jiang\textsuperscript{1,2}

\textsuperscript{1}Shanghai Key Lab of Intelligent Information Processing, School of Computer Science, Fudan University
\textsuperscript{2}Shanghai Collaborative Innovation Center on Intelligent Visual Computing
\textsuperscript{3}Department of Computer Science and Information Technology, La Trobe University

\{jgzu20,zhengwang17,chengjingjing,ygj\}@fudan.edu.cn, phoebe.chen@latrobe.edu.au

Abstract

Real-world data typically follow a long-tailed distribution, where a few majority categories occupy most of the data while most minority categories contain a limited number of samples. Classification models minimizing cross-entropy struggle to represent and classify the tail classes. Although the problem of learning unbiased classifiers has been well studied, methods for representing imbalanced data are under-explored. In this paper, we focus on representation learning for imbalanced data. Recently, supervised contrastive learning has shown promising performance on balanced data recently. However, through our theoretical analysis, we find that for long-tailed data, it fails to form a regular simplex which is an ideal geometric configuration for representation learning. To correct the optimization behavior of SCL and further improve the performance of long-tailed visual recognition, we propose a novel loss for balanced contrastive learning (BCL). Compared with SCL, we have two improvements in BCL: class-averaging, which balances the gradient contribution of negative classes; class-complement, which allows all classes to appear in every mini-batch. The proposed balanced contrastive learning (BCL) method satisfies the condition of forming a regular simplex and assists the optimization of cross-entropy. Equipped with BCL, the proposed two-branch framework can obtain a stronger feature representation and achieve competitive performance on long-tailed benchmark datasets such as CIFAR-10-LT, CIFAR-100-LT, ImageNet-LT, and iNaturalist2018.

1. Introduction

Deep neural networks have achieved remarkable success in a series of computer vision tasks, such as image recognition [4,12,25], video analysis [37,49], object detection [35], etc. These achievements are owing largely to the availability of large-scale dataset such as ImageNet [11], where each class has sufficient and equal amount of training samples. However, real-world datasets are often imbalanced, where many classes have only a few samples and few classes have a great number of samples. Deep models trained with such unbalanced data usually generalize badly on balanced testing data, especially for rare classes. Improving recognition performance with unbalanced data poses a huge challenge to modern deep learning methods.

To tackle the problem of learning with imbalanced data, early methods mainly focus on re-sampling the training data [1,2,14,33] or re-weighting the loss functions [9,21,45] to pay more attention to rare classes. Recently, diverse methods have emerged. For example, Logit compensation methods [3,31,38] calibrate distribution between the training data and the test data. Decoupling [23] adopts a two-stage training scheme where the classifier is re-
balanced in the second stage. The work in [44] has multiple distribution-aware experts for responding to samples of different class frequencies. Nevertheless, contrastive learning approaches are less explored before, not until contrastive learning [8, 22, 42] are introduced. We attach great importance to representation learning because it’s the most remarkable capability of deep models.

In this paper, we focus on using supervised contrastive learning (SCL) [24] to assist representation learning. Supervised contrastive loss has achieved better performance than supervised cross-entropy loss on large-scale classification problems. The work in [16] then has explained in detail the reason for the excellent performance of SCL on the balanced datasets. Despite the great success, some recent work [8, 22] indicate that high-frequency classes dominate SCL for representing imbalanced data, which results in unsatisfactory performance across all classes. To analyze the optimizing behavior of SCL in learning the representations for long-tailed data, we depict the geometric arrangement of representations of training instances when the lower bound of loss is achieved. Specifically, we decouple the lower bound of loss by deriving two competing dynamics: an attraction term and a repulsion term as in [16]. We reveal that the long-tailed distribution mainly affects the repulsion term. At the minimal supervised contrastive loss, the representations of classes of long-tailed data no longer attain a regular simplex configuration. In other words, when all instances with the same label collapse to points, these points are not equidistant from each other. A regular simplex configuration empirically confers important benefits, such as better generalization performance [32]. Besides, it has been proved to be the target geometric configuration of SCL on balanced data [16], hence forming a regular simplex configuration will benefit recognition on long-tailed data [15].

Inspired by our analysis, we urge the model learning on imbalanced data to form a regular simplex, and propose a balanced contrastive learning (BCL) method (illustrated in Fig. 1). We have two modifications in BCL that distinguish it from SCL. First, class-complement introduces the class-center embeddings, i.e., prototypes, as instances for comparison in every mini-batch. Second, class-averaging has the gradient contributions of all negatives of each class averaged for every mini-batch. Through these two improvements, BCL ensures that the overall lower bound of the loss is a class-independent constant, and alleviates the imbalance problem of SCL when representing long-tailed data. Furthermore, We adopt a cross-entropy loss with logit compensation to obtain a balanced classifier. Logit compensation can effectively alleviate overlooking tail classes in the classifier learning [3, 20, 31, 34]. Overall, we propose a two-branch framework to implement the mentioned techniques, i.e., a contrastive learning branch with BCL and a classification branch with logit compensated cross-entropy.

Our main contributions are as follows:

- We present a theoretical analysis showing that supervised contrastive learning forms an undesired asymmetric geometry configuration for long-tailed data due to the overwhelming numerical dominance of the head classes.

- Motivated by our analysis, we extend supervised contrastive learning to balanced contrastive learning, which overcomes the imbalance problem and remains a regular simplex configuration of long-tailed data.

- The proposed two-branch framework combines the classification module and the balanced contrastive learning module, achieving competitive results on several popular long-tailed datasets.

2. Related Work

**Long-tailed Recognition** Early solutions to address the long-tailed problem comprise two main ideas: re-sampling and re-weighting. Re-sampling methods undersample [1, 14] high-frequency classes or oversample [1, 2, 33] low-frequency classes. Re-weighting methods [9, 21, 45] assign different losses to different training samples for each class or each example. Both BBN [51] and Decoupling [23] indicate that the re-balancing method is detrimental to representation learning. BBN dynamically adjusts the weights between features from the instance-balanced sampling branch and the reversed sampling branch. While Decoupling proposes a two-stage learning strategy that firstly obtains a good feature extractor and secondly fixes the feature extractor and fine-tunes the classifier. Recently proposed logit compensation methods [3, 23, 31, 38] learn relatively larger margins between different classes based on the prior of class frequencies. For example, logit adjustment [31] derives the general form of the compensation value based on the optimal Bayesian classifier. Our proposed framework simultaneously improves the representation learning with BCL and strengthens the classifier learning with logit compensation in an end-to-end manner.

**Supervised Contrastive Learning** Contrastive learning (CL) trains the model in a pairwise way by aggregating semantically similar samples while excluding semantically dissimilar ones, which has been employed for feature representation learning in various tasks [17, 43, 46]. SimCLR [5] and MoCo [18] are two typical types of self-supervised contrastive learning. SCL [24] leverages label information for fully-supervised representation learning, leading to state-of-the-art performance for image classification.

**Contrastive Learning for Long-tailed Recognition** Trained on long-tailed data, conventional contrastive learning can pose potential problems. SSP [48] boosts long-tailed learning with self-supervised and semi-supervised
contrastive learning. Hybrid-SC [42] designs a two-branch network, using a supervised contrastive learning branch for learning better representations and a classifier branch for eliminating the bias of the classifier towards head classes. Despite our framework shares a similar two-branch architecture with Hybrid-SC, our framework differs from Hybrid-SC as we introduce BCL loss in the SCL branch for dealing with the domination of majority classes problem. Hybrid-PSC [42] is proposed to overcome the memory bottleneck problem of SCL with a prototype for each class for contrast. PaCo [8] overcomes the performance degradation of SCL by introducing a set of class-wise learnable centers. KCL [22] adopts the two-stage learning paradigm and uses the same number of positives for all classes in every batch. The recently proposed TSC [28] is the most relevant work to ours, which urges the features of classes closer to the target classes and reduces the distances between low-frequency classes to similar samples, and a small temperature tends to be less tolerant to similar samples [41]. Note that we omit \( \tau \) in the following contrastive losses for simplicity.

Similar to [16], we also introduce the class-specific batch-wise loss:

\[
L_{SCL}(Z; Y, B, y) = \begin{cases} 
\sum_{i \in B_y} L_i & \text{if } |B_y| > 1 \\
0 & \text{else}
\end{cases}
\]  

(2)

Regular simplex. A set of points \( \zeta_1, \ldots, \zeta_K \in \mathbb{R}^h \) form the vertices of a regular simplex inscribed in the hypersphere of radius \( \rho > 0 \), if and only if the following conditions hold:

1. \( \sum_{i \in [K]} \zeta_i = 0 \)
2. \( ||\zeta_i|| = \rho \), for \( i \in [K] \)
3. \( \exists d \in \mathbb{R} : d = \langle \zeta_i, \zeta_j \rangle \) for \( 1 \leq i < j \leq K \)

where \( h, K \in \mathbb{N} \) with \( K \leq h + 1 \), and \( \langle \cdot \rangle \) stands for the inner product operation. Regular simplex has a highly symmetric structure that all vertices are equally spaced. Given a balanced dataset, it’s worth mentioning that when supervised contrastive loss attain the minimum, representations of each class collapse to the vertices of a regular simplex spontaneously [16,32]. See the illustration in Fig. 2(a).

3.2. Analysis

Drawbacks of SCL. To clearly show the optimization behaviour of SCL on long-tailed data, we mainly focus on the variation of the geometry configuration formed by the representations of each class. Although representations of each class collapse to the vertices of a regular simplex when supervised contrastive loss attain its minimum on a balanced dataset, SCL forms an asymmetrical configuration on long-tailed data as shown in Fig. 2(b). In the following, we will give an in-depth analysis on the loss function to show why the geometry configuration changes for imbalanced data. In particular, we analyze the lower-bound of the loss. Since directly computing the lower bound on the whole long-tailed dataset is often intractable, we pay attention to the loss of a specific mini-batch instead.

\[ \text{Theorem 1. Assuming the normalization function is applied for feature embedding, let } Z = (z_1, \ldots, z_N) \in \mathbb{Z}^N \text{ be an } N \text{ point configuration with labels } Y = (y_1, \ldots, y_N) \in [K]^N, \text{ where } Z = \{z \in \mathbb{R}^h : ||z|| = 1\}. \text{ The class-specific batch-} \]
Supervised contrastive learning (SCL) aims to maximize the similarity between a sample $z$ and its anchor sample $y$ defined as $\log(1 + \exp(z_i \cdot y_j)), \quad \forall i, j \in B_B$. However, when samples are imbalanced, the head classes dominate the repulsion term and the representation may collapse to the head classes. We introduce a loss function $L_{SC}$ for class-specific batch-wise learning:

$$L_{SC}(Z; Y, B, y) \geq \sum_{i \in B_y} \log((|B_y| - 1) + |B_y| \exp \left( \sum_{k \in B_Y} z_i \cdot z_k - \frac{1}{|B_y| - 1} \sum_{j \in B_y \setminus \{i\}} z_i \cdot z_j \right))$$

for every $y \in \mathcal{Y}_B$, where $\mathcal{Y}_B$ denotes the set of classes that appear within batch $B$. Theorem 1 states that the class-specific batch-wise loss after performing class-averaging is bounded by:

$$L_{BC}(Z; Y, B, y) \geq \sum_{i \in B_y} \log(1 + (|B_y| - 1) \times \exp \left( \sum_{q \in \mathcal{Y}_B \setminus \{y\}} \frac{1}{|B_q|} \sum_{k \in B_q} z_i \cdot z_k - \frac{1}{|B_y| - 1} \sum_{j \in B_y \setminus \{i\}} z_i \cdot z_j \right))$$

for every $y \in \mathcal{Y}_B$, where $\mathcal{Y}_B = \mathcal{Y}$ for every $B$, the overall loss is given by:

$$L_{BC}(Z; Y) \geq |\mathcal{D}| \log(1 + (K - 1) \exp(-\frac{K}{K-1}))$$

where $\mathcal{D}$ represents the dataset. Here, the normalization term is neglected. Recall that $Z$ is an $N$ point configuration with labels $Y$, the equality of Eq. 5 is attained if and only if the following conditions hold:}

1. $\forall n \in [N] : z_n = \zeta y_n$
2. $\zeta_1, \ldots, \zeta_K$ form a regular simplex

Proof. See the Supplementary Material.

Note that condition (1) implies variability collapse, and condition (2) demonstrates the regular simplex structure.

When balanced contrastive loss attains its lower bound,
Figure 4. Overview of the proposed framework. The framework consists of a classification branch and a balanced contrastive learning branch. \(v_2\) and \(v_3\) adopt the same augmentation method different from \(v_1\). The backbone is shared between two branches. The classifier weights are separately transformed by a MLP to be used as prototypes. All representations are \(\ell_2\)-normalized for balanced contrastive loss.

each negative class contributes to the gradient equally. Additionally, BCL ensures that the loss of each sample is consistent and class-independent when attaining its lower bound, which implies that the learning will be less biased towards the head classes (Fig. 2(c)).

3.3. Balanced Contrastive Learning

Class-averaging The key idea is to average the instances of each class in a mini-batch so that each class has an approximate contribution for optimizing. Intuitively, it reduces the proportion of head classes in the denominator and emphasizes the importance of tails. In Section 3.2, we take the loss form as \(L_1\) for analysis. However, there are other ways to implement class-averaging. Here, we give three loss forms as follows:

\[ L_1 = -\frac{1}{|B_y| - 1} \sum_{p \in B_y \setminus \{i\}} \log \frac{\exp(z_i \cdot z_p)}{\sum_{j \in Y_B} \sum_{k \in B_j} \exp(z_i \cdot z_k)} \]  
\[ L_2 = -\frac{1}{|B_y| - 1} \sum_{p \in B_y \setminus \{i\}} \log \frac{\exp(z_i \cdot z_p)}{\sum_{j \in Y_B} \exp(z_i \cdot z_p) \sum_{k \in B_j} z_i \cdot z_k} \]  
\[ L_3 = -\log \frac{\exp(z_i \cdot z_{c_j})}{\sum_{j \in Y_B} \exp(z_i \cdot z_{c_j})} \]  

where the \(|B_j|\) term will minus one when the positive class is averaged. The only difference between our \(L_1\) and \(L_2\) is that the averaging operation takes place in different positions. \(L_1\) performs averaging outside the exponential function, while \(L_2\) has averaging inside the exponential function. Since \(-\log\) and \(\exp\) are convex functions, which implies that \(L_1 \geq L_2\) by Jensen’s inequality. \(L_3\) take the form proposed in other prototype-based contrastive learning methods [26, 36, 42], where each sample is pulled towards its class prototype and pushed away from others. Note that \(z \in Z \subseteq \mathbb{R}^k\) is \(\ell_2\)-normalized for the inner product, so that \(||z||_2 = 1\). A comparison of their performance is in the experiment section. Below, we choose \(L_1\) for optimization.

Class-complement To have all classes appear in every mini-batch, we introduce class-center representations, i.e., prototypes for balanced contrastive learning. Now we have the formulation of balanced contrastive loss as follows:

\[ \mathcal{L}_{BCL} = -\frac{1}{|B_y|} \times \sum_{p \in B_y \setminus \{i\}} \log \frac{\exp(z_i \cdot z_p)}{\sum_{j \in Y_B} \sum_{k \in B_j \cup \{c_j\}} \exp(z_i \cdot z_k)} \]  

where \(c_j\) is the index of the prototype. In practice, we perform a nonlinear mapping of the classifier weights and regard the output as the prototype of each class.

By applying both class-averaging and class-complement, the lower bound is a class-independent constant, avoiding the model’s preference for head classes. Note that in practice, we perform class-complement before class-averaging, as formulated in Eq. 9 and illustrated in Fig. 3.

Framework The overview of the proposed framework is shown in Fig. 4. It consists of two main components: a classification branch and a contrastive learning branch. Both branches are trained simultaneously and share the same feature extractor. BCL is a unified end-to-end model, which is different from conventional contrastive learning methods that follow a two-stage training strategy. We have different augmentation methods for the two branches. Three different views are generated in total, where \(v_1\) is the view used for the classification task, \(v_2\) and \(v_3\) are the pairwise views for
the contrastive learning task. Following the work in [5, 24], we utilize a symmetric architecture for the contrastive learning branch. We use a MLP with one hidden layer to obtain the representation $z_i$ for contrastive learning where $z_i = W_1 \sigma(W_2 f_i)$ and $\sigma$ is a ReLU function. Instead of using mean embeddings [36] or learnable parameters [8, 42] as prototypes, we are motivated by that the weights of the linear classifier are co-linear with these simplex vertices to which the classes collapse [16, 32]. Specifically, we have class-specific weights $w_1, w_2, \ldots, w_K$ after a non-linear transformation MLP as prototypes $z_{c_1}, z_{c_2}, \ldots, z_{c_K}$. Ablations of using different configurations of views and different forms of prototypes are given in the Supplementary Material. All representations for contrastive learning is $\ell_2$ normalized to ensure the feature space is a unit hypersphere.

**Optimization with Logit Compensation** For long-tailed learning tasks, due to the imbalance of data, the output logit of the last classification layer usually exists bias. Logit compensation aims to eliminate the bias caused by the imbalance of data and learn the rectification of the boundary [3, 31, 38]. The compensation can be applied during either training or testing. Previous work [3, 9, 20, 31, 38] illustrates the desirability of logit compensation in long-tailed visual tasks and it can be summarized as the following form

$$L_{LC}(y, \varphi(x)) = -\alpha_y \log \frac{\exp(\varphi_y(x) + \delta_y)}{\sum_{y' \in [Y]} \exp(\varphi_{y'}(x) + \delta_{y'})}$$

(10)

Here, $\alpha_y$ is the factor that controls the importance of class $y$, $\delta_y$ is the compensation for class $y$ and its value is related to class-frequency. We define $\alpha_y = 1$, $\delta_y = \log P_y$ as in [31] and perform logit compensation along with training, where $P_y$ denotes the class prior of label $y$.

Finally, we have the following loss for training:

$$L = \lambda L_{LC} + \mu L_{BCL}$$

(11)

where $\lambda$ and $\mu$ are hyperparameters that control the impact of $L_{LC}$ and $L_{BCL}$, respectively. In addition, the contrastive branch only intends for the backbone to learn the desired feature embeddings.

### 4. Experiment

#### 4.1. Dataset

**Long-Tailed CIFAR-10 and CIFAR-100** CIFAR-10-LT and CIFAR-100-LT are the subsets of CIFAR-10 and CIFAR-100, respectively. Both CIFAR-10 and CIFAR-100 contain 50,000 images for training and 10,000 images for the validation of size $32 \times 32$ with 10 and 100 classes respectively. Following [3, 9, 51], we use the same long-tailed version for a fair comparison. The imbalanced factor $\beta$ is defined by $\beta = N_{\text{max}} / N_{\text{min}}$, and this reflects the degree of imbalance in the data. The imbalance factors used in the experiment are set to 100, 50, and 10.

**ImageNet-LT** ImageNet-LT is proposed in [30], which is a long-tailed version of vanilla ImageNet by sampling a subset following the Pareto distribution with power value $\alpha = 0.6$. It consists of 115.8K images of 1000 classes in total with 1280 to 5 images per class.

**iNaturalist 2018** iNaturalist 2018 [40] is a large-scale dataset containing 437.5K images from 8,142 classes. It is long-tailed by nature with an extremely imbalanced distribution. In addition to long-tailed recognition, this dataset is also used for evaluating the fine-grained classification task.

#### 4.2. Implementation details

For both CIFAR-10-LT and CIFAR-100-LT, we use the ResNet-32 as the backbone. Same to [8], we use AutoAugment [6] and Cutout [13] as data augmentation strategies for the classification branch and SimAugment [5] for the contrastive learning branch. To control the impact of $L_{LC}$ and $L_{BCL}$, $\lambda$ is set to 2.0, $\mu$ is 0.6, and the temperature $\tau$ is set to 0.1. We set the batch size as 256 and the weight decay as $5e^{-4}$. The dimension of the hidden layer and the output layer of MLP are set to 512 and 128, respectively. We run BCL for 200 epochs with the learning rate warms up to 0.15 within the first 5 epochs and decays at epoch 160 and 180 with a step size of 0.1. Following [8], we also run the model for 400 epochs, where the learning rate warms up to 0.15 within the first 10 epochs and decays at epoch 360 and 380 with a step size of 0.1. We train the above models with one Nvidia GeForce 1080Ti GPU.

For ImageNet-LT, we use ResNet-50 [19] and ResNeXt-50-32x4d [47] as our backbone. We run BCL for 90 epochs with an initial learning rate of 0.1 and the weight decay is $5e^{-4}$. For iNaturalist, we use ResNet-50 as our backbone.
Table 3. Top-1 accuracy of ResNet-32 on CIFAR-100-LT and CIFAR-10-LT. The best results are marked in bold. † denotes results borrowed from [42]. We report the results of 200 epochs.

<table>
<thead>
<tr>
<th>Method</th>
<th>CIFAR-100-LT</th>
<th>CIFAR-10-LT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Imbalance Factor</td>
<td></td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>50</td>
</tr>
<tr>
<td>SSP [48]</td>
<td>43.43</td>
<td>41.71</td>
</tr>
<tr>
<td>Focal loss† [29]</td>
<td>38.41</td>
<td>44.32</td>
</tr>
<tr>
<td>CB-Focal [10]</td>
<td>39.60</td>
<td>45.17</td>
</tr>
<tr>
<td>BBN [51]</td>
<td>42.56</td>
<td>47.02</td>
</tr>
<tr>
<td>Casual model [39]</td>
<td>44.10</td>
<td>50.30</td>
</tr>
<tr>
<td>LDAM-DRW [3]</td>
<td>42.04</td>
<td>46.62</td>
</tr>
<tr>
<td>ResLT [7]</td>
<td>48.21</td>
<td>52.71</td>
</tr>
<tr>
<td>Hybrid-SC [42]</td>
<td>46.72</td>
<td>51.87</td>
</tr>
<tr>
<td>MetaSAug-LDAM [27]</td>
<td>48.01</td>
<td>52.27</td>
</tr>
<tr>
<td>BCL (ours)</td>
<td>51.93</td>
<td>56.59</td>
</tr>
</tbody>
</table>

Table 4. Top-1 accuracy of ResNet-32 on CIFAR-100-LT with an imbalance factor of 100. We report the results of 200 epochs and 400 epochs. † and ‡ denote results borrowed from [36] and [8].

and run BCL for 100 epochs using an initial learning rate of 0.2 and the weight decay is 1e-4. For both ImageNet-LT and iNaturalist 2018, we use cosine scheduling for learning rate, \( \lambda \) is set to 1.0 and \( \mu \) is set to 0.35. The batch size is set to 256. We use the RandAug augmentation strategy for the classification branch, and SimAug for the contrastive learning branch. The performances of different augmentation strategies are in the Supplementary Material. To reduce memory consumption, the dimension of the output layer of MLP is set to 1024 for both datasets. We use the cosine classifier. All models are trained using SGD optimizer with a momentum set to 0.9. For a fair comparison, we reproduce PaCo of ResNext-50 on ImageNet-LT for 180 epochs.

4.3. Ablation study

We perform several ablation studies to characterize the proposed BCL method. All experiments are performed on CIFAR-100 with an imbalance factor of 100. First, we compare the performances of different class-averaging implementations (i.e., \( L_1 \), \( L_2 \) and \( L_3 \)) mentioned in Section 3.3. The main difference between \( L_1 \) and \( L_2 \) is the order in which the averaging operations are implemented. For \( L_3 \), we use the prototype implemented in our work instead of the average of all embeddings of the same class. As shown in Table 1, \( L_1 \) achieves the best performance, which is consistent with our previous analysis. Surprisingly, \( L_3 \) achieves better performance than \( L_2 \), which may be attributed to the well-represented characteristics of prototypes.

To demonstrate the superiority of the balanced contrastive loss, we compare the performance of the primary components of the loss in Table 2. We use the cross-entropy loss with logit compensation (LC) as the vanilla baseline. SC denotes a baseline that adds the contrastive learning branch with conventional supervised contrastive loss. Class-complement and class-averaging are the main techniques of the proposed balanced contrastive loss. We show that using either class-complement or class-averaging alone cannot improve the overall accuracy. In contrast, a significant performance boost can be obtained when both of them are applied, which indicates both components are indispensable components to achieve stronger performance.

4.4. Main results

Long-tailed CIFAR The comparison results between the proposed BCL and other existing methods on long-tailed CIFAR are shown in Table 3. As can be seen from the table, BCL consistently outperforms the other methods. Furthermore, BCL achieves better performance on long-tailed CIFAR datasets with large imbalance factors. We note that the accuracy gap between BCL and Hybrid-SC decreases as the degree of data imbalance reduces. This result is mainly attributed to the fact that the conventional supervised contrastive loss leads to more serious bias in representation learning when imbalance problem is more severe.

Further, we report the accuracy on three groups of classes, including Many-shot(>100 images), Medium-shot(2~100 images), Few-shot(<20 images), on CIFAR-100-LT with imbalance factor as 100. Additionally, for a
fair comparison with [8], we report the accuracy on 200 and 400 epochs. DRO-LT [36] is also a contrastive learning method which extends prototypical contrastive learning by introducing distributional robustness. As shown in Table 4, BCL surpasses DRO-LT by 4.6 and PaCo by 1.9 with 200 training epochs and 400 training epochs, respectively. It’s worthwhile to mention that different to most of the previous methods which compromise the performance of the head classes, our BCL further improves the performance of the head classes while simultaneously improving the tails.

**ImageNet-LT** Table 5 and Table 6 list the results on ImageNet-LT. We report the overall Top-1 accuracy as well as the Top-1 accuracy on Many-shot, Medium-shot, and Few-shot groups. Compared with Balanced Softmax [34], which proposes a logit compensation by adjusting the predictions according to the class frequency. BCL significantly outperforms Balanced Softmax on all groups, confirming the well-learned representations can boost the overall performance. LWS [23], τ-norm [23], and DisAlign [50] adopt the two-stage learning strategy. These methods focus on fine-tuning the classifier in the second stage, while they neglect the bias implied in the representation learning stage. PaCo [8] uses a set of parametric centers in supervised contrastive learning. These centers are assigned with a much greater weight, which can be regarded as the weights of a classifier. However, the prototypes used in BCL complement the samples of each class to make sure that all classes appear in every mini-batch. Compared with PaCo, BCL achieves a better overall accuracy of 57.1% with remarkable accuracy improvements on the head and few classes.

**iNaturalist 2018** Table 5 shows the experimental results on iNaturalist 2018. Since BCL is a contrastive learning method, it benefits more from a longer training time. However, for a fair comparison, we report the results of various models trained for up to 100 epochs. Hybrid-SC [42] and Hybrid-PSC [42] are contrastive learning approaches, and their performances are inferior to BCL due to the potential bias produced in representation learning. RIDE [44] trains model from an ensemble-based manner. RIDE obtains high performance by routing diverse experts. However, it’s different from our motivation of boosting long-tailed learning via training a sufficient encoder and a balanced feature space. Compared with the ensemble-based model RIDE, BCL consistently demonstrates better recognition performance and achieves the overall accuracy at 71.8%.

### Table 5. Top-1 accuracy of ResNet-50 on ImageNet-LT and iNaturalist 2018. All methods are trained for up to 100 epochs.

<table>
<thead>
<tr>
<th>Methods</th>
<th>ImageNet-LT</th>
<th>iNaturalist 2018</th>
</tr>
</thead>
<tbody>
<tr>
<td>τ-norm [23]</td>
<td>46.7</td>
<td>65.6</td>
</tr>
<tr>
<td>cRT [23]</td>
<td>49.6</td>
<td>65.2</td>
</tr>
<tr>
<td>LWS [23]</td>
<td>49.9</td>
<td>65.9</td>
</tr>
<tr>
<td>BBN [51]</td>
<td>-</td>
<td>66.3</td>
</tr>
<tr>
<td>Hybrid-SC [42]</td>
<td>-</td>
<td>66.7</td>
</tr>
<tr>
<td>Hybrid-PSC [42]</td>
<td>-</td>
<td>68.1</td>
</tr>
<tr>
<td>SSP [48]</td>
<td>51.3</td>
<td>68.1</td>
</tr>
<tr>
<td>KCL [22]</td>
<td>51.5</td>
<td>68.6</td>
</tr>
<tr>
<td>DisAlign [50]</td>
<td>52.9</td>
<td>69.5</td>
</tr>
<tr>
<td>RIDE(2 experts) [44]</td>
<td>54.4</td>
<td>71.4</td>
</tr>
<tr>
<td>BCL(Ours)</td>
<td><strong>56.0</strong></td>
<td><strong>71.8</strong></td>
</tr>
</tbody>
</table>

### Table 6. Top-1 accuracy of ResNext-50 on ImageNet-LT. We report the results of 90 epochs and 180 epochs. † denotes results reproduced using the code released by authors, both with RandAug.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Many</th>
<th>Medium</th>
<th>Few</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>90 epochs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Focal Loss† [29]</td>
<td>64.3</td>
<td>37.1</td>
<td>8.2</td>
<td>43.7</td>
</tr>
<tr>
<td>τ-norm [23]</td>
<td>59.1</td>
<td>46.9</td>
<td>30.7</td>
<td>49.4</td>
</tr>
<tr>
<td>Balanced Softmax† [34]</td>
<td>62.2</td>
<td>48.8</td>
<td>29.8</td>
<td>51.4</td>
</tr>
<tr>
<td>LWS [23]</td>
<td>60.2</td>
<td>47.2</td>
<td>30.3</td>
<td>49.9</td>
</tr>
<tr>
<td>LADE [20]</td>
<td>62.3</td>
<td>49.3</td>
<td>31.2</td>
<td>51.9</td>
</tr>
<tr>
<td>Casual model [39]</td>
<td>62.7</td>
<td>48.8</td>
<td>31.6</td>
<td>51.8</td>
</tr>
<tr>
<td>DisAlign [50]</td>
<td>62.7</td>
<td>52.1</td>
<td>31.4</td>
<td>53.4</td>
</tr>
<tr>
<td>RIDE(2 experts) [44]</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>55.9</td>
</tr>
<tr>
<td>BCL(Ours)</td>
<td><strong>67.2</strong></td>
<td><strong>53.9</strong></td>
<td><strong>36.5</strong></td>
<td><strong>56.7</strong></td>
</tr>
<tr>
<td>180 epochs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LADE [20]</td>
<td>65.1</td>
<td>48.9</td>
<td>33.4</td>
<td>53.0</td>
</tr>
<tr>
<td>Balanced Softmax†</td>
<td>65.8</td>
<td>53.2</td>
<td>34.1</td>
<td>55.4</td>
</tr>
<tr>
<td>PaCo† [8]</td>
<td>64.4</td>
<td><strong>55.7</strong></td>
<td>33.7</td>
<td>56.0</td>
</tr>
<tr>
<td>BCL(Ours)</td>
<td><strong>67.9</strong></td>
<td>54.2</td>
<td><strong>36.6</strong></td>
<td><strong>57.1</strong></td>
</tr>
</tbody>
</table>

5. **Conclusion**

In this work, we investigated the problem of long-tailed recognition from the perspective of representation learning. We provided in-depth analysis to demonstrate that existing supervised contrastive learning forms an undesired asymmetric geometry configuration for long-tailed data. To tackle the imbalanced data representation learning problem, we developed a balanced contrastive loss, so that all classes are optimized for a regular simplex configuration that yields a balanced feature space. In addition to BCL, we employ a classification branch with logit compensation to tackle the biased classifier. Overall we have presented a framework unifying both branches. We conducted extensive experiments on the long-tailed benchmarks of long-tailed CIFAR, ImageNet-LT, and iNaturalist 2018. The experimental results adequately demonstrate the superiority of BCL compared to existing long-tailed learning methods.

6. **Acknowledgements**

This work was supported in part by NSFC project (#62072116), Shanghai Municipal Commission of Economy and Informatization Project (2020-GYHLW-01009), and in part by Shanghai Pujiang Program (20PJ1401900).
References

ings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5212–5221, 2021. 7


[38] Jiajun Tang, Jianqiang Huang, and Hanwang Zhang. Long-tailed classification by keeping the good and removing the bad momentum causal effect. Advances in Neural Information Processing Systems, 33, 2020. 7, 8


