Class-Balanced Loss Based on Effective Number of Samples

Yin Cui[1,2*] Menglin Jia[1] Tsung-Yi Lin[3] Yang Song[4] Serge Belongie[1,2]
[1]Cornell University [2]Cornell Tech [3]Google Brain [4]Alphabet Inc.

Abstract

*With the rapid increase of large-scale, real-world datasets, it becomes critical to address the problem of long-tailed data distribution (*i.e.*, a few classes account for most of the data, while most classes are under-represented). Existing solutions typically adopt class re-balancing strategies such as re-sampling and re-weighting based on the number of observations for each class. In this work, we argue that as the number of samples increases, the additional benefit of a newly added data point will diminish. We introduce a novel theoretical framework to measure data overlap by associating with each sample a small neighboring region rather than a single point. The effective number of samples is defined as the volume of samples and can be calculated by a simple formula $(1-\beta^n)/(1-\beta)$, where n is the number of samples and $\beta \in [0, 1)$ is a hyperparameter. We design a re-weighting scheme that uses the effective number of samples for each class to re-balance the loss, thereby yielding a* class-balanced loss. *Comprehensive experiments are conducted on artificially induced long-tailed CIFAR datasets and large-scale datasets including ImageNet and iNaturalist. Our results show that when trained with the proposed class-balanced loss, the network is able to achieve significant performance gains on long-tailed datasets.*

1. Introduction

The recent success of deep Convolutional Neural Networks (CNNs) for visual recognition [25, 36, 37, 16] owes much to the availability of large-scale, real-world annotated datasets [7, 27, 48, 40]. In contrast with commonly used visual recognition datasets (*e.g.*, CIFAR [24, 39], ImageNet ILSVRC 2012 [7, 33] and CUB-200 Birds [42]) that exhibit roughly uniform distributions of class labels, real-world datasets have skewed [21] distributions, with a *long-tail*: a few dominant classes claim most of the examples, while most of the other classes are represented by relatively few examples. CNNs trained on such data perform poorly for weakly represented classes [19, 15, 41, 4].

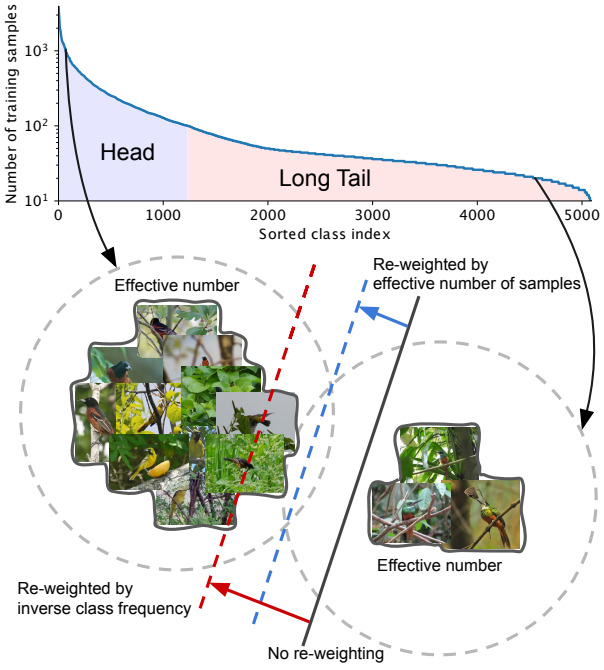

Figure 1. Two classes, one from the head and one from the tail of a long-tailed dataset (iNaturalist 2017 [40] in this example), have drastically different number of samples. Models trained on these samples are biased toward dominant classes (black solid line). Re-weighing the loss by inverse class frequency usually yields poor performance (red dashed line) on real-world data with high class imbalance. We propose a theoretical framework to quantify the effective number of samples by taking data overlap into consideration. A class-balanced term is designed to re-weight the loss by inverse effective number of samples. We show in experiments that the performance of a model can be improved when trained with the proposed class-balanced loss (blue dashed line).

A number of recent studies have aimed to alleviate the challenge of long-tailed training data [3, 31, 17, 41, 43, 12, 47, 44]. In general, there are two strategies: re-sampling and cost-sensitive re-weighting. In re-sampling, the number of examples is directly adjusted by over-sampling (adding repetitive data) for the minor class or under-sampling (removing data) for the major class, or both. In cost-sensitive re-weighting, we influence the loss function by assigning

*The work was performed while Yin Cui and Yang Song worked at Google (a subsidiary of Alphabet Inc.).

relatively higher costs to examples from minor classes. In the context of deep feature representation learning using CNNs, re-sampling may either introduce large amounts of duplicated samples, which slows down the training and makes the model susceptible to overfitting when over-sampling, or discard valuable examples that are important for feature learning when under-sampling. Due to these disadvantages of applying re-sampling for CNN training, the present work focuses on re-weighting approaches, namely, how to design a better class-balanced loss.

Typically, a class-balanced loss assigns sample weights inversely proportionally to the class frequency. This simple heuristic method has been widely adopted [17, 43]. However, recent work on training from large-scale, real-world, long-tailed datasets [30, 28] reveals poor performance when using this strategy. Instead, they use a "smoothed" version of weights that are empirically set to be inversely proportional to the square root of class frequency. These observations suggest an interesting question: how can we design a better class-balanced loss that is applicable to a diverse array of datasets?

We aim to answer this question from the perspective of sample size. As illustrated in Figure 1, we consider training a model to discriminate between a major class and a minor class from a long-tailed dataset. Due to highly imbalanced data, directly training the model or re-weighting the loss by inverse number of samples cannot yield satisfactory performance. Intuitively, the more data, the better. However, since there is information overlap among data, as the number of samples increases, the marginal benefit a model can extract from the data diminishes. In light of this, we propose a novel theoretical framework to characterize data overlap and calculate the effective number of samples in a model- and loss-agnostic manner. A class-balanced re-weighting term that is inversely proportional to the effective number of samples is added to the loss function. Extensive experimental results indicate that this class-balanced term provides a significant boost to the performance of commonly used loss functions for training CNNs on long-tailed datasets.

Our key contributions can be summarized as follows: (1) We provide a theoretical framework to study the effective number of samples and show how to design a class-balanced term to deal with long-tailed training data. (2) We show that significant performance improvements can be achieved by adding the proposed class-balanced term to existing commonly used loss functions including softmax cross-entropy, sigmoid cross-entropy and focal loss. In addition, we show our class-balanced loss can be used as a generic loss for visual recognition by outperforming commonly-used softmax cross-entropy loss on ILSVRC 2012. We believe our study on quantifying the effective number of samples and class-balanced loss can offer useful guidelines for researchers working in domains with long-tailed class distributions.

2. Related Work

Most of previous efforts on long-tailed imbalanced data can be divided into two regimes: re-sampling [35, 12, 4, 50] (including over-sampling and under-sampling) and cost-sensitive learning [38, 49, 17, 22, 34].

Re-Sampling. Over-sampling adds repeated samples from minor classes, which could cause the model to overfit. To solve this, novel samples can be either interpolated from neighboring samples [5] or synthesized [14, 50] for minor classes. However, the model is still error-prone due to noise in the novel samples. It was argued that even if over-sampling incurs risks from removing important samples, under-sampling is still preferred over over-sampling [9].

Cost-Sensitive Learning. Cost-Sensitive Learning can be traced back to a classical method in statistics called importance sampling [20], where weights are assigned to samples in order to match a given data distribution. Elkan *et al*. [10] studied how to assign weights to adjust the decision boundary to match a given target in the case of binary classification. For imbalanced datasets, weighting by inverse class frequency [17, 43] or a smoothed version of inverse square root of class frequency [30, 28] are often adopted. As a generalization of smoothed weighting with a theoretically grounded framework, we focus on (a) how to quantify the effective number of samples and (b) using it to re-weight the loss. Another line of important work aims to study sample difficulty in terms of loss and assign higher weights to hard examples [11, 29, 8, 26]. Samples from minor classes tend to have higher losses than those from major classes as the features learned in minor classes are usually poorer. However, there is no direct connection between sample difficulty and the number of samples. A side effect of assigning higher weights to hard examples is the focus on harmful samples (*e.g*., noisy data or mislabeled data) [23, 32]. In our work, we do not make any assumptions on the sample difficulty and data distribution. By improving the focal loss [26] using our class-balanced term in experiments, we show that our method is complementary to re-weighting based on sample difficulty.

It is noteworthy to mention that previous work has also explored other ways of dealing with data imbalance, including transferring the knowledge learned from major classes to minor classes [3, 31, 43, 6, 44] and designing a better training objective via metric learning [17, 47, 45].

Covering and Effective Sample Size. Our theoretical framework is inspired by the random covering problem [18], where the goal is to cover a large set by a sequence of i.i.d. random small sets. We simplify the problem in Section 3 by making reasonable assumptions. Note that the effective number of samples proposed in this paper is different from the concept of effective sample size in statistics. The effective sample size is used to calculate variance when samples are correlated.

3. Effective Number of Samples

We formulate the data sampling process as a simplified version of random covering. The key idea is to associate each sample with a small neighboring region instead of a single point. We present our theoretical framework and the formulation of calculating effective number of samples.

3.1. Data Sampling as Random Covering

Given a class, denote the set of all possible data in the feature space of this class as $\mathcal{S}$. We assume the volume of $\mathcal{S}$ is N and $N \geq 1$. Denote each data as a subset of $\mathcal{S}$ that has the unit volume of 1 and may overlap with other data. Consider the data sampling process as a random covering problem where each data (subset) is randomly sampled from $\mathcal{S}$ to cover the entire set of $\mathcal{S}$. The more data is being sampled, the better the coverage of $\mathcal{S}$ is. The expected total volume of sampled data increases as the number of data increases and is bounded by N. Therefore, we define:

Definition 1 (Effective Number)**.** The *effective number* of samples is the expected volume of samples.

The calculation of the expected volume of samples is a very difficult problem that depends on the shape of the sample and the dimensionality of the feature space [18]. To make the problem tamable, we simplify the problem by not considering the situation of partial overlapping. That is, we assume a newly sampled data can only interact with previously sampled data in two ways: either entirely inside the set of previously sampled data with the probability of p or entirely outside with the probability of $1-p$, as illustrated in Figure 2. As the number of sampled data points increases, the probability p will be higher.

Before we dive into the mathematical formulations, we discuss the connection between our definition of effective number of samples and real-world visual data. Our idea is to capture the diminishing marginal benefits by using more data points of a class. Due to intrinsic similarities among real-world data, as the number of samples grows, it is highly possible that a newly added sample is a near-duplicate of existing samples. In addition, CNNs are trained with heavy data augmentations, where simple transformations such as random cropping, re-scaling and horizontal flipping will be applied to the input data. In this case, all augmented examples are also considered as same with the original example. Presumably, the stronger the data augmentation is, the smaller the N will be. The small neighboring region of a sample is a way to capture all near-duplicates and instances that can be obtained by data augmentation. For a class, N can be viewed as the number of *unique prototypes*.

3.2. Mathematical Formulation

Denote the effective number (expected volume) of samples as E_n, where $n \in \mathbb{Z}_{>0}$ is the number of samples.

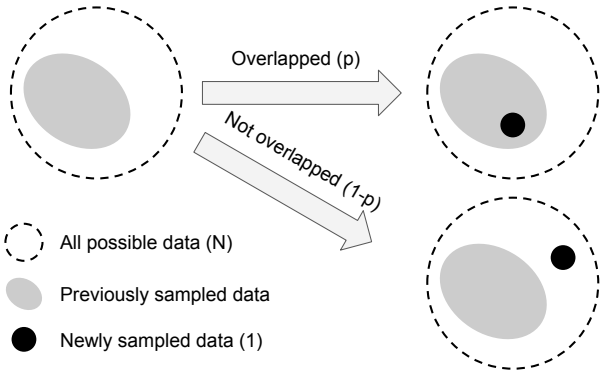

Figure 2. Giving the set of all possible data with volume N and the set of previously sampled data, a new sample with volume 1 has the probability of p being overlapped with previous data and the probability of $1-p$ not being overlapped.

Proposition 1 (Effective Number)**.** $E_n = (1-\beta^n)/(1-\beta)$, where $\beta = (N-1)/N$.

Proof. We prove the proposition by induction. It is obvious that $E_1 = 1$ because there is no overlapping. So $E_1 = (1-\beta^1)/(1-\beta) = 1$ holds. Now let's consider a general case where we have previously sampled $n-1$ examples and are about to sample the n^{th} example. Now the expected volume of previously sampled data is E_{n-1} and the newly sampled data point has the probability of $p = E_{n-1}/N$ to be overlapped with previous samples. Therefore, the expected volume after sampling n^{th} example is:

$$E_n = pE_{n-1} + (1-p)(E_{n-1}+1) = 1 + \frac{N-1}{N}E_{n-1}. \quad (1)$$

Assume $E_{n-1} = (1-\beta^{n-1})/(1-\beta)$ holds, then

$$E_n = 1 + \beta\frac{1-\beta^{n-1}}{1-\beta} = \frac{1-\beta+\beta-\beta^n}{1-\beta} = \frac{1-\beta^n}{1-\beta}. \quad (2)$$

□

The above proposition shows that the effective number of samples is an exponential function of n. The hyperparameter $\beta \in [0, 1)$ controls how fast E_n grows as n increases.

Another explanation of the effective number E_n is:

$$E_n = (1-\beta^n)/(1-\beta) = \sum_{j=1}^{n} \beta^{j-1}. \quad (3)$$

This means that the j^{th} sample contributes β^{j-1} to the effective number. The total volume N for all possible data in the class can then be calculated as:

$$N = \lim_{n\to\infty} \sum_{j=1}^{n} \beta^{j-1} = 1/(1-\beta). \quad (4)$$

This is consistent with our definition of β in the proposition.

Implication 1 (Asymptotic Properties). $E_n = 1$ if $\beta = 0$ ($N = 1$). $E_n \to n$ as $\beta \to 1$ ($N \to \infty$).

Proof. If $\beta = 0$, then $E_n = (1 - 0^n)/(1 - 0) = 1$. In the case of $\beta \to 1$, denote $f(\beta) = 1 - \beta^n$ and $g(\beta) = 1 - \beta$. Since $\lim_{\beta \to 1} f(\beta) = \lim_{\beta \to 1} g(\beta) = 0$, $g'(\beta) = -1 \neq 0$ and $\lim_{\beta \to 1} f'(\beta)/g'(\beta) = \lim_{\beta \to 1}(-n\beta^{n-1})/(-1) = n$ exists, using L'Hôpital's rule, we have

$$\lim_{\beta \to 1} E_n = \lim_{\beta \to 1} \frac{f(\beta)}{g(\beta)} = \lim_{\beta \to 1} \frac{f'(\beta)}{g'(\beta)} = n. \tag{5}$$

□

The asymptotic property of E_n shows that when N is large, the effective number of samples is same as the number of samples n. In this scenario, we think the number of unique prototypes N is large, thus there is no data overlap and every sample is unique. On the other extreme, if $N = 1$, this means that we believe there exist a single prototype so that all the data in this class can be represented by this prototype via data augmentation, transformations, *etc*.

4. Class-Balanced Loss

The *Class-Balanced Loss* is designed to address the problem of training from imbalanced data by introducing a weighting factor that is inversely proportional to the effective number of samples. The class-balanced loss term can be applied to a wide range of deep networks and loss functions.

For an input sample $\mathbf{x}$ with label $y \in \{1, 2, \ldots, C\}$ [1], where C is the total number of classes, suppose the model's estimated class probabilities are $\mathbf{p} = [p_1, p_2, \ldots, p_C]^\top$, where $p_i \in [0, 1]\ \forall\ i$, we denote the loss as $\mathcal{L}(\mathbf{p}, y)$. Suppose the number of samples for class i is n_i, based on Equation 2, the proposed effective number of samples for class i is $E_{n_i} = (1 - \beta_i^{n_i})/(1 - \beta_i)$, where $\beta_i = (N_i - 1)/N_i$. Without further information of data for each class, it is difficult to empirically find a set of good hyperparameters N_i for all classes. Therefore, in practice, we assume N_i is only dataset-dependent and set $N_i = N$, $\beta_i = \beta = (N - 1)/N$ for all classes in a dataset.

To balance the loss, we introduce a weighting factor α_i that is inversely proportional to the effective number of samples for class i: $\alpha_i \propto 1/E_{n_i}$. To make the total loss roughly in the same scale when applying α_i, we normalize α_i so that $\sum_{i=1}^{C} \alpha_i = C$. For simplicity, we abuse the notation of $1/E_{n_i}$ to denote the normalized weighting factor in the rest of our paper.

Formally speaking, given a sample from class i that contains n_i samples in total, we propose to add a weighting factor $(1 - \beta)/(1 - \beta^{n_i})$ to the loss function, with hyperparameter $\beta \in [0, 1)$. The class-balanced (CB) loss can be written as:

$$\text{CB}(\mathbf{p}, y) = \frac{1}{E_{n_y}} \mathcal{L}(\mathbf{p}, y) = \frac{1 - \beta}{1 - \beta^{n_y}} \mathcal{L}(\mathbf{p}, y), \tag{6}$$

where n_y is the number of samples in the ground-truth class y. We visualize class-balanced loss in Figure 3 as a function of n_y for different β. Note that $\beta = 0$ corresponds to no re-weighting and $\beta \to 1$ corresponds to re-weighing by inverse class frequency. The proposed novel concept of effective number of samples enables us to use a hyperparameter β to smoothly adjust the class-balanced term between no re-weighting and re-weighing by inverse class frequency.

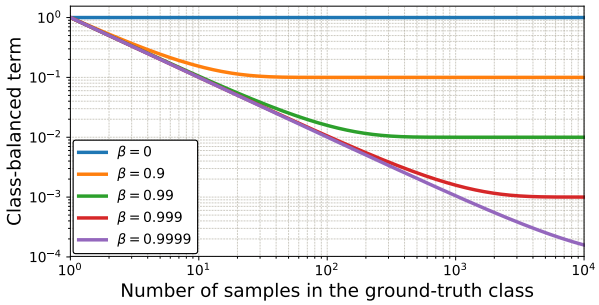

Figure 3. Visualization of the proposed class-balanced term $(1 - \beta)/(1 - \beta^{n_y})$, where n_y is the number of samples in the ground-truth class. Both axes are in log scale. For a long-tailed dataset where major classes have significantly more samples than minor classes, setting β properly re-balances the relative loss across classes and reduces the drastic imbalance of re-weighing by inverse class frequency.

The proposed class-balanced term is model-agnostic and loss-agnostic in the sense that it's independent to the choice of loss function $\mathcal{L}$ and predicted class probabilities $\mathbf{p}$. To demonstrate the proposed class-balanced loss is generic, we show how to apply class-balanced term to three commonly used loss functions: softmax cross-entropy loss, sigmoid cross-entropy loss and focal loss.

4.1. Class-Balanced Softmax Cross-Entropy Loss

Suppose the predicted output from the model for all classes are $\mathbf{z} = [z_1, z_2, \ldots, z_C]^\top$, where C is the total number of classes. The softmax function regards each class as mutual exclusive and calculate the probability distribution over all classes as $p_i = \exp(z_i)/\sum_{j=1}^{C} \exp(z_j), \forall\ i \in \{1, 2, \ldots, C\}$. Given a sample with class label y, the softmax cross-entropy (CE) loss for this sample is written as:

$$\text{CE}_{\text{softmax}}(\mathbf{z}, y) = -\log\left(\frac{\exp(z_y)}{\sum_{j=1}^{C} \exp(z_j)}\right). \tag{7}$$

[1] For simplicity, we derive the loss function by assuming there is only one ground-truth label for a sample.

Suppose class y has n_y training samples, the class-balanced (CB) softmax cross-entropy loss is:

$$\text{CB}_{\text{softmax}}(\mathbf{z}, y) = -\frac{1-\beta}{1-\beta^{n_y}} \log\left(\frac{\exp(z_y)}{\sum_{j=1}^{C}\exp(z_j)}\right). \quad (8)$$

4.2. Class-Balanced Sigmoid Cross-Entropy Loss

Different from softmax, class-probabilities calculated by sigmoid function assume each class is independent and not mutually exclusive. When using sigmoid function, we regard multi-class visual recognition as multiple binary classification tasks, where each output node of the network is performing a one-vs-all classification to predict the probability of the target class over the rest of classes. Compared with softmax, sigmoid presumably has two advantages for real-world datasets: (1) Sigmoid doesn't assume the mutual exclusiveness among classes, which aligns well with real-world data, where a few classes might be very similar to each other, especially in the case of large number of fine-grained classes. (2) Since each class is considered independent and has its own predictor, sigmoid unifies single-label classification with multi-label prediction. This is a nice property to have since real-world data often has more than one semantic label.

Using same notations as softmax cross-entropy, for simplicity, we define z_i^t as:

$$z_i^t = \begin{cases} z_i, & \text{if } i = y. \\ -z_i, & \text{otherwise.} \end{cases} \quad (9)$$

Then the sigmoid cross-entropy (CE) loss can be written as:

$$\begin{aligned} \text{CE}_{\text{sigmoid}}(\mathbf{z}, y) &= -\sum_{i=1}^{C} \log\left(\text{sigmoid}(z_i^t)\right) \\ &= -\sum_{i=1}^{C} \log\left(\frac{1}{1+\exp(-z_i^t)}\right). \end{aligned} \quad (10)$$

The class-balanced (CB) sigmoid cross-entropy loss is:

$$\text{CB}_{\text{sigmoid}}(\mathbf{z}, y) = -\frac{1-\beta}{1-\beta^{n_y}} \sum_{i=1}^{C} \log\left(\frac{1}{1+\exp(-z_i^t)}\right). \quad (11)$$

4.3. Class-Balanced Focal Loss

The recently proposed focal loss (FL) [26] adds a modulating factor to the sigmoid cross-entropy loss to reduce the relative loss for well-classified samples and focus on difficult samples. Denote $p_i^t = \text{sigmoid}(z_i^t) = 1/(1+\exp(-z_i^t))$, the focal loss can be written as:

$$\text{FL}(\mathbf{z}, y) = -\sum_{i=1}^{C}(1-p_i^t)^{\gamma}\ \log(p_i^t). \quad (12)$$

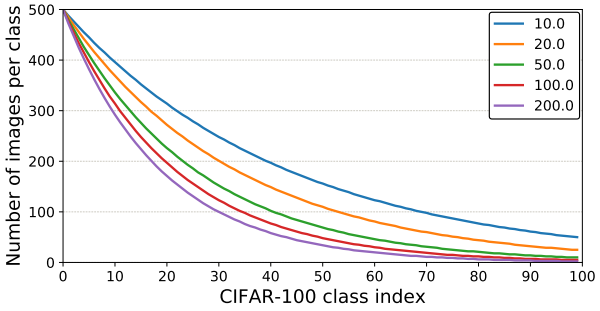

Figure 4. Number of training samples per class in artificially created long-tailed CIFAR-100 datasets with different imbalance factors.

Dataset Name	# Classes	Imbalance
Long-Tailed CIFAR-10	10	10.00 - 200.00
Long-Tailed CIFAR-100	100	10.00 - 200.00
iNaturalist 2017	5,089	435.44
iNaturalist 2018	8,142	500.00
ILSVRC 2012	1,000	1.78

Table 1. Datasets that are used to evaluate the effectiveness of class-balanced loss. We created 5 long-tailed versions of both CIFAR-10 and CIFAR-100 with imbalance factors of 10, 20, 50, 100 and 200 respectively.

The class-balanced (CB) focal loss is:

$$\text{CB}_{\text{focal}}(\mathbf{z}, y) = -\frac{1-\beta}{1-\beta^{n_y}} \sum_{i=1}^{C}(1-p_i^t)^{\gamma}\ \log(p_i^t). \quad (13)$$

The original focal loss has an α-balanced variant. The class-balanced focal loss is same as α-balanced focal loss when $\alpha_t = (1-\beta)/(1-\beta^{n_y})$. Therefore, the class-balanced term can be viewed as an explicit way to set α_t in focal loss based on the effective number of samples.

5. Experiments

The proposed class-balanced losses are evaluated on artificially created long-tailed CIFAR [24] datasets with controllable degrees of data imbalance and real-world long-tailed datasets iNaturalist 2017 [40] and 2018 [1]. To demonstrate our loss is generic for visual recognition, we also present experiments on ImageNet data (ILSVRC 2012 [33]). We use deep residual networks (ResNet) [16] with various depths and train all networks from scratch.

5.1. Datasets

Long-Tailed CIFAR. To analyze the proposed class-balanced loss, long-tailed versions of CIFAR [24] are created by reducing the number of training samples per class according to an exponential function $n = n_i\mu^i$, where i

Dataset Name	Long-Tailed CIFAR-10						Long-Tailed CIFAR-100					
Imbalance	200	100	50	20	10	1	200	100	50	20	10	1
Softmax	**34.32**	29.64	25.19	17.77	13.61	6.61	65.16	61.68	56.15	48.86	44.29	29.07
Sigmoid	34.51	**29.55**	23.84	**16.40**	**12.97**	**6.36**	64.39	**61.22**	55.85	48.57	44.73	**28.39**
Focal ($\gamma = 0.5$)	36.00	29.77	**23.28**	17.11	13.19	6.75	65.00	61.31	55.88	48.90	44.30	28.55
Focal ($\gamma = 1.0$)	34.71	29.62	23.29	17.24	13.34	6.60	**64.38**	61.59	**55.68**	**48.05**	**44.22**	28.85
Focal ($\gamma = 2.0$)	35.12	30.41	23.48	16.77	13.68	6.61	65.25	61.61	56.30	48.98	45.00	28.52
Class-Balanced	**31.11**	**25.43**	**20.73**	**15.64**	**12.51**	**6.36***	**63.77**	**60.40**	**54.68**	**47.41**	**42.01**	**28.39***
Loss Type	SM	Focal	Focal	SM	SGM	SGM	Focal	Focal	SGM	Focal	Focal	SGM
β	0.9999	0.9999	0.9999	0.9999	0.9999	-	0.9	0.9	0.99	0.99	0.999	-
γ	-	1.0	2.0	-	-	-	1.0	1.0	-	0.5	0.5	-

Table 2. Classification error rate of ResNet-32 trained with different loss functions on long-tailed CIFAR-10 and CIFAR-100. We show best results of class-balanced loss with best hyperparameters (SM represents Softmax and SGM represents Sigmoid) chosen via cross-validation. Class-balanced loss is able to achieve significant performance gains. $*$ denotes the case when each class has same number of samples, class-balanced term is always 1 therefore it reduces to the original loss function.

is the class index (0-indexed), n_i is the original number of training images and $\mu \in (0, 1)$. The test set remains unchanged. We define the imbalance factor of a dataset as the number of training samples in the largest class divided by the smallest. Figure 4 shows number of training images per class on long-tailed CIFAR-100 with imbalance factors ranging from 10 to 200. We conduct experiments on long-tailed CIFAR-10 and CIFAR-100.

iNaturalist. The recently introduced iNaturalist species classification and detection dataset [40] is a real-world long-tailed dataset containing 579,184 training images from 5,089 classes in its 2017 version and 437,513 training images from 8,142 classes in its 2018 version [1]. We use the official training and validation splits in our experiments.

ImageNet. We use the ILSVRC 2012 [33] split containing 1,281,167 training and 50,000 validation images.

Table 1 summarizes all datasets used in our experiments along with their imbalance factors.

5.2. Implementation

Training with sigmoid-based losses. Conventional training scheme of deep networks initializes the last linear classification layer with bias $b = 0$. As pointed out by Lin *et al*. [26], this could cause instability of training when using sigmoid function to get class probabilities. This is because using $b = 0$ with sigmoid function in the last layer induces huge loss at the beginning of the training as the output probability for each class is close to 0.5. Therefore, for training with sigmoid cross-entropy loss and focal loss, we assume the class prior is $\pi = 1/C$ for each class, where C is the number of classes, and initialize the bias of the last layer as $b = -\log\left((1-\pi)/\pi\right)$. In addition, we remove the L_2 regularization (weight decay) for the bias b of the last layer.

We used Tensorflow [2] to implement and train all the models by stochastic gradient descent with momentum. We trained residual networks with 32 layers (ResNet-32) to conduct all experiments on CIFAR. Similar to Zagoruyko *et al*. [46], we noticed a disturbing effect in training ResNets on CIFAR that both loss and validation error gradually went up after the learning rate drop, especially in the case of high data imbalance. We found that setting learning rate decay to 0.01 instead of 0.1 solved the problem. Models on CIFAR were trained with batch size of 128 on a single NVIDIA Titan X GPU for 200 epochs. The initial learning rate was set to 0.1, which was then decayed by 0.01 at 160 epochs and again at 180 epochs. We also used linear warm-up of learning rate [13] in the first 5 epochs. On iNaturalist and ILSVRC 2012 data, we followed the same training strategy used by Goyal *et al*. [13] and trained residual networks with batch size of 1024 on a single cloud TPU. Since the scale of focal loss is smaller than softmax and sigmoid cross-entropy loss, when training with focal loss, we used $2\times$ and $4\times$ larger learning rate on ILSVRC 2012 and iNaturalist respectively. Code, data and pre-trained models are available at: https://github.com/richardaecn/class-balanced-loss.

5.3. Visual Recognition on Long-Tailed CIFAR

We conduct extensive studies on long-tailed CIFAR datasets with various imbalance factors. Table 2 shows the performance of ResNet-32 in terms of classification error rate on the test set. We present results of using softmax cross-entropy loss, sigmoid cross-entroy loss, focal loss with different γ, and the proposed class-balanced loss with best hyperparameters chosen via cross-validation. The search space of hyperparameters is {softmax, sigmoid, focal} for loss type, $\beta \in \{0.9, 0.99, 0.999, 0.9999\}$ (Section 4), and $\gamma \in \{0.5, 1.0, 2.0\}$ for focal loss [26].

From results in Table 2, we have the following observations: (1) With properly selected hyperparameters, class-balanced loss is able to significantly improve the performance of commonly used loss functions on long-tailed datasets. (2) Softmax cross-entropy is overwelmingly used as the loss function for visual recognition tasks. How-

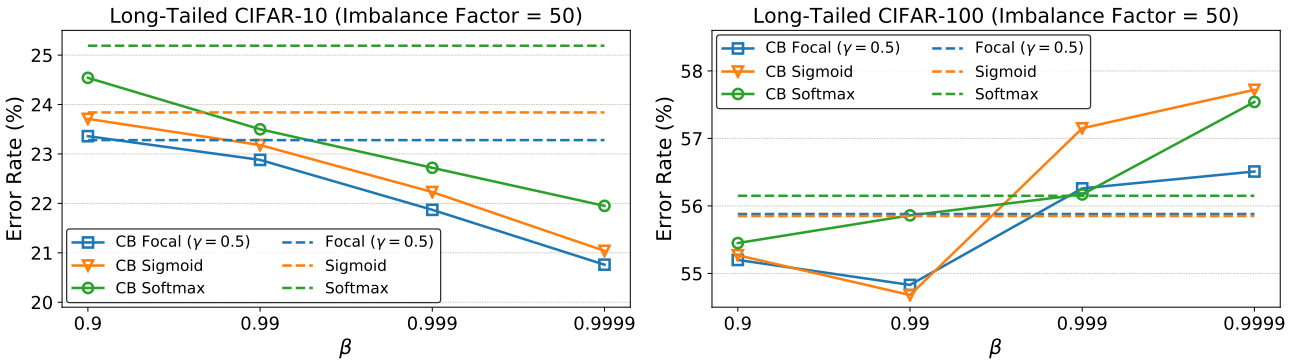

Figure 5. Classification error rate when trained with and without the class-balanced term. On CIFAR-10, class-balanced loss yields consistent improvement across different β and the larger the β is, the larger the improvement is. On CIFAR-100, $\beta = 0.99$ or $\beta = 0.999$ improves the original loss, whereas a larger β hurts the performance.

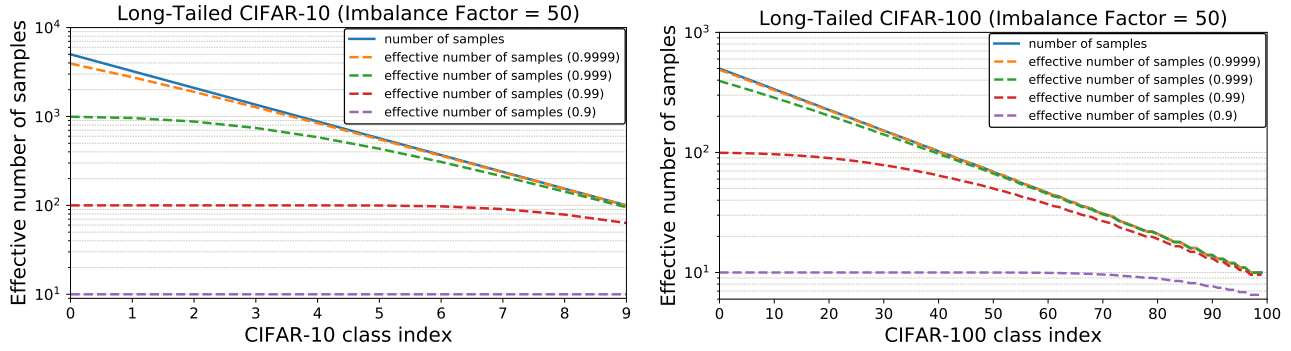

Figure 6. Effective number of samples with different β on long-tailed CIFAR-10 and CIFAR-100 with the imbalance of 50. This is a semi-log plot with vertical axis in log scale. When $\beta \to 1$, effective number of samples is same as number of samples. When β is small, effective number of samples are similar across all classes.

ever, following the training strategy in Section 5.2, sigmoid cross-entropy and focal loss are able to outperform softmax cross-entropy in most cases. (3) The best β is 0.9999 on CIFAR-10 unanimously. But on CIFAR-100, datasets with different imbalance factors tend to have different and smaller optimal β.

To understand the role of β and class-balanced loss better, we use the long-tailed dataset with imbalance factor of 50 as an example to show the error rate of the model when trained with and without the class-balanced term in Figure 5. Interestingly, for CIFAR-10, class-balanced term always improves the performance of the original loss and more performance gain can be obtained with larger β. However, on CIFAR-100, only small values of β improve the performance, whereas larger values degrade the performance. Figure 6 illustrates the effective number of samples under different β. On CIFAR-10, when re-weighting based on $\beta = 0.9999$, the effective number of samples is close to the number of samples. This means the best re-weighting strategy on CIFAR-10 is similar with re-weighting by inverse class frequency. On CIFAR-100, the poor performance of using larger β suggests that re-weighting by inverse class frequency is not a wise choice. Instead, we need to use a smaller β that has smoother weights across classes. This is reasonable because $\beta = (N-1)/N$, so larger β means larger N. As discussed in Section 3, N can be interpreted as the number of unique prototypes. A fine-grained dataset should have a smaller N compared with a coarse-grained one. For example, the number of unique prototypes of a specific bird species should be smaller than the number of unique prototypes of a generic bird class. Since classes in CIFAR-100 are more fine-grained than CIFAR-10, CIFAR-100 should have smaller N compared with CIFAR-10. This explains our observations on the effect of β.

5.4. Visual Recognition on Large-Scale Datasets

To demonstrate the proposed class-balanced loss can be used on large-scale real-world datasets, we present results of training ResNets with different depths on iNaturalist 2017, iNaturalist 2018 and ILSVRC 2012.

					iNaturalist 2017		iNaturalist 2018		ILSVRC 2012	
Network	Loss	β	γ	Input Size	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
ResNet-50	Softmax	-	-	224×224	45.38	22.67	42.86	21.31	23.92	7.03
ResNet-101	Softmax	-	-	224×224	42.57	20.42	39.47	18.86	22.65	6.47
ResNet-152	Softmax	-	-	224×224	41.42	19.47	38.61	18.07	21.68	5.92
ResNet-50	CB Focal	0.999	0.5	224×224	41.92	20.92	38.88	18.97	22.71	6.72
ResNet-101	CB Focal	0.999	0.5	224×224	39.06	18.96	36.12	17.18	21.57	5.91
ResNet-152	CB Focal	0.999	0.5	224×224	38.06	18.42	35.21	16.34	20.87	5.61
ResNet-50	CB Focal	0.999	0.5	320×320	38.16	18.28	35.84	16.85	21.99	6.27
ResNet-101	CB Focal	0.999	0.5	320×320	34.96	15.90	32.02	14.27	20.25	5.34
ResNet-152	CB Focal	0.999	0.5	320×320	33.73	14.96	30.95	13.54	19.72	4.97

Table 3. Classification error rate on large-scale datasets trained with different loss functions. The proposed class-balanced term combined with focal loss (CB Focal) is able to outperform softmax cross-entropy by a large margin.

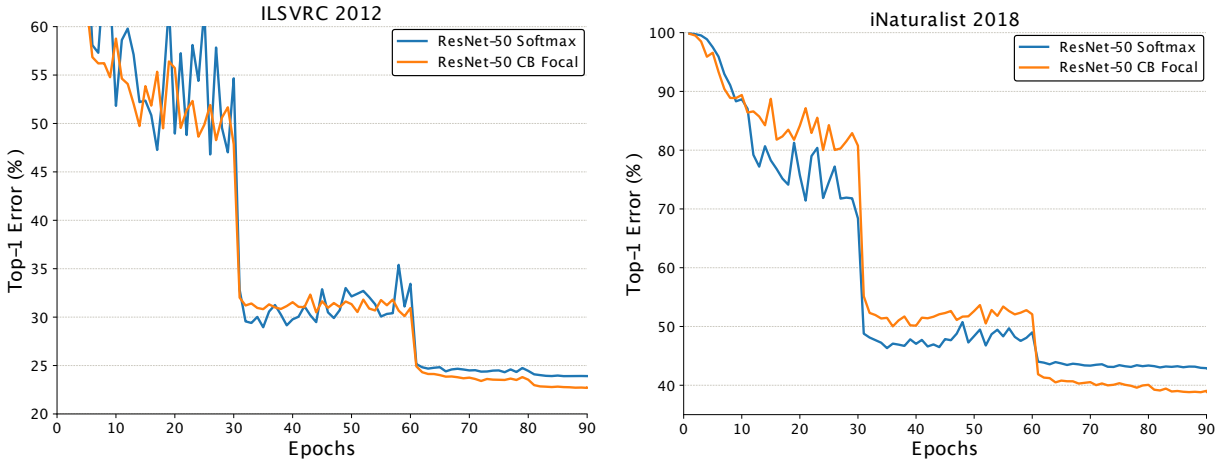

Figure 7. Training curves of ResNet-50 on ILSVRC 2012 (left) and iNaturalist 2018 (right). Class-balanced focal loss with $\beta = 0.999$ and $\gamma = 0.5$ outperforms softmax cross-entropy after 60 epochs.

Table 3 summarizes the top-1 and top-5 error rate on the validation set of all datasets. We use the class-balanced focal loss since it has more flexibility and find $\beta = 0.999$ and $\gamma = 0.5$ yield reasonably good performance on all datasets. From results we can see that we are able to outperform commonly used softmax cross-entropy loss on ILSVRC 2012, and by large margins on iNaturalist. Notably, ResNet-50 is able to achieve comparable performance with ResNet-152 on iNaturalist and ResNet-101 on ILSVRC 2012 when using class-balanced focal loss to replace softmax cross-entropy loss. Training curves on ILSVRC 2012 and iNaturalist 2018 are shown in Figure 7. Class-balanced focal loss starts to show its advantage after 60 epochs of training.

6. Conclusion and Discussion

In this work, we have presented a theoretically sounded framework to address the problem of long-tailed distribution of training data. The key idea is to take data overlap into consideration to help quantify the effective number of samples. Following this framework, we further propose a *class-balanced loss* to re-weight loss inversely with the effective number of samples per class. Extensive studies on artificially induced long-tailed CIFAR datasets have been conducted to understand and analyze the proposed loss. The benefit of the class-balanced loss has been verified by experiments on both CIFAR and large-scale datasets including iNaturalist and ImageNet.

Our proposed framework provides a non-parametric means of quantifying data overlap, since we don't make any assumptions about the data distribution. This makes our loss generally applicable to a wide range of existing models and loss functions. Intuitively, a better estimation of the effective number of samples could be obtained if we know the data distribution. In the future, we plan to extend our framework by incorporating reasonable assumptions on the data distribution or designing learning-based, adaptive methods.

Acknowledgment. This work was supported in part by a Google Focused Research Award.

Appendix A: More Experimental Results

We present more comprehensive experimental results in this appendix.

Visual Recognition on Long-Tailed CIFAR. On long-tailed CIFAR datasets with imbalance factors of 200, 100, 50, 20 and 10, we trained ResNet-32 models [16] using softmax loss (SM), sigmoid loss (SGM) and focal loss with both the original loss and class-balanced variants with $\beta = 0.9$, 0.99, 0.999 and 0.9999. For focal loss, we used $\gamma = 0.5$, 1.0 and 2.0. In addition to long-tailed CIFAR-10 and CIFAR-100 datasets mentioned in Section 5.1 of the main paper, we also conduct experiments on CIFAR-20 dataset, which has same images as CIFAR-100 dataset but annotated with 20 coarse-grained class-labels [24].

Classification error rates on long-tailed CIFAR-10, CIFAR-20 and CIFAR-100 datasets are shown in Figure 8, Figure 9 and Figure 10 respectively. Each row in the figure corresponds to the model trained with a specific loss function, in the form of $\{\text{loss name}\}\_\{\gamma\}\_\{\beta\}$, and each column corresponds to a long-tailed dataset with specific imbalance factor. We color-code each column to visualize results, with lighter colors represent lower error rates and darker colors for higher error rates. Note that results in each column of Table 2 in the main paper are classification error rates using the original losses and the best-performed class-balanced loss that is same as the lowest error rates in the corresponding column of Figure 8 and Figure 10 (marked by underline). From these results, we can see that in general, higher β yields better performance on CIFAR-10. However, on CIFAR-20 and CIFAR-100, lower β is needed to achieve good performance, suggesting that we cannot directly re-weight the loss by inverse class-frequency, but to re-weight based on the effective number of samples. These results support the analysis in Section 5.3 of our main paper.

Figure 8. Classification error rates of ResNet-32 models trained with different loss functions on long-tailed CIFAR-10.

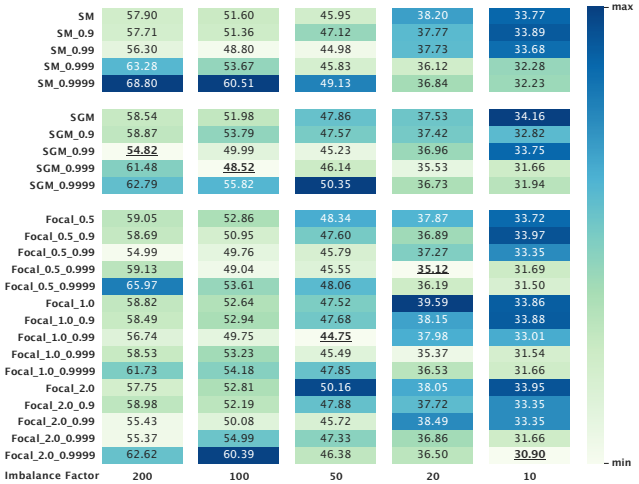

Figure 9. Classification error rates of ResNet-32 models trained with different loss functions on long-tailed CIFAR-20 (CIFAR-100 with 20 coarse-grained class-labels).

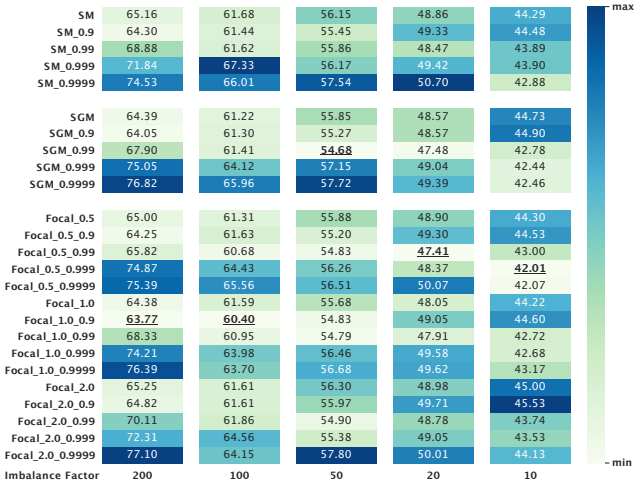

Figure 10. Classification error rates of ResNet-32 models trained with different loss functions on long-tailed CIFAR-100.

References

[1] The iNaturalist 2018 Competition Dataset. https://github.com/visipedia/inat_comp. 5, 6

[2] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: A system for large-scale machine learning. In *OSDI*, 2016. 6

[3] S. Bengio. Sharing representations for long tail computer vision problems. In *ICMI*, 2015. 1, 2

[4] M. Buda, A. Maki, and M. A. Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 2018. 1, 2

[5] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: synthetic minority over-sampling technique. *JAIR*, 2002. 2

[6] Y. Cui, Y. Song, C. Sun, A. Howard, and S. Belongie. Large scale fine-grained categorization and domain-specific transfer learning. In *CVPR*, 2018. 2

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 1

[8] Q. Dong, S. Gong, and X. Zhu. Class rectification hard mining for imbalanced deep learning. In *ICCV*, 2017. 2

[9] C. Drummond, R. C. Holte, et al. C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In *ICML Workshop*, 2003. 2

[10] C. Elkan. The foundations of cost-sensitive learning. In *IJCAI*, 2001. 2

[11] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 1997. 2

[12] Y. Geifman and R. El-Yaniv. Deep active learning over the long tail. *arXiv preprint arXiv:1711.00941*, 2017. 1, 2

[13] P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, and K. He. Accurate, large minibatch sgd: training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017. 6

[14] H. He, Y. Bai, E. A. Garcia, and S. Li. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *IEEE International Joint Conference on Neural Networks*, 2008. 2

[15] H. He and E. A. Garcia. Learning from imbalanced data. *IEEE Transactions on Knowledge & Data Engineering*, 2008. 1

[16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 5, 9

[17] C. Huang, Y. Li, C. Change Loy, and X. Tang. Learning deep representation for imbalanced classification. In *CVPR*, 2016. 1, 2

[18] S. Janson. Random coverings in several dimensions. *Acta Mathematica*, 1986. 2, 3

[19] N. Japkowicz and S. Stephen. The class imbalance problem: A systematic study. *Intelligent data analysis*, 2002. 1

[20] H. Kahn and A. W. Marshall. Methods of reducing sample size in monte carlo computations. *Journal of the Operations Research Society of America*, 1953. 2

[21] M. G. Kendall et al. The advanced theory of statistics. *The advanced theory of statistics.*, (2nd Ed), 1946. 1

[22] S. H. Khan, M. Hayat, M. Bennamoun, F. A. Sohel, and R. Togneri. Cost-sensitive learning of deep feature representations from imbalanced data. *IEEE transactions on neural networks and learning systems*, 2018. 2

[23] P. W. Koh and P. Liang. Understanding black-box predictions via influence functions. In *ICML*, 2017. 2

[24] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009. 1, 5, 9

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Neural Information Processing Systems*, 2012. 1

[26] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. *PAMI*, 2018. 2, 5, 6

[27] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1

[28] D. Mahajan, R. Girshick, V. Ramanathan, K. He, M. Paluri, Y. Li, A. Bharambe, and L. van der Maaten. Exploring the limits of weakly supervised pretraining. In *ECCV*, 2018. 2

[29] T. Malisiewicz, A. Gupta, and A. A. Efros. Ensemble of exemplar-svms for object detection and beyond. In *ICCV*, 2011. 2

[30] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Neural Information Processing Systems*, 2013. 2

[31] W. Ouyang, X. Wang, C. Zhang, and X. Yang. Factors in finetuning deep model for object detection with long-tail distribution. In *CVPR*, 2016. 1, 2

[32] M. Ren, W. Zeng, B. Yang, and R. Urtasun. Learning to reweight examples for robust deep learning. In *ICML*, 2018. 2

[33] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015. 1, 5, 6

[34] N. Sarafianos, X. Xu, and I. A. Kakadiaris. Deep imbalanced attribute classification using visual attention aggregation. In *ECCV*, 2018. 2

[35] L. Shen, Z. Lin, and Q. Huang. Relay backpropagation for effective learning of deep convolutional neural networks. In *ECCV*, 2016. 2

[36] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[37] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 1

[38] K. M. Ting. A comparative study of cost-sensitive boosting algorithms. In *ICML*, 2000. 2

[39] A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *PAMI*, 2008. 1

[40] G. Van Horn, O. Mac Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, and S. Belongie. The inaturalist species classification and detection dataset. In *CVPR*, 2018. 1, 5, 6

[41] G. Van Horn and P. Perona. The devil is in the tails: Fine-grained classification in the wild. *arXiv preprint arXiv:1709.01450*, 2017. 1

[42] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. *California Institute of Technology*, 2011. 1

[43] Y.-X. Wang, D. Ramanan, and M. Hebert. Learning to model the tail. In *Neural Information Processing Systems*, 2017. 1, 2

[44] X. Yin, X. Yu, K. Sohn, X. Liu, and M. Chandraker. Feature transfer learning for deep face recognition with long-tail data. *arXiv preprint arXiv:1803.09014*, 2018. 1, 2

[45] C. You, C. Li, D. P. Robinson, and R. Vidal. A scalable exemplar-based subspace clustering algorithm for class-imbalanced data. In *European Conference on Computer Vision*, 2018. 2

[46] S. Zagoruyko and N. Komodakis. Wide residual networks. In *BMVC*, 2016. 6

[47] X. Zhang, Z. Fang, Y. Wen, Z. Li, and Y. Qiao. Range loss for deep face recognition with long-tailed training data. In *CVPR*, 2017. 1, 2

[48] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba. Places: A 10 million image database for scene recognition. *PAMI*, 2017. 1

[49] Z.-H. Zhou and X.-Y. Liu. Training cost-sensitive neural networks with methods addressing the class imbalance problem. *IEEE Transactions on Knowledge and Data Engineering*, 2006. 2

[50] Y. Zou, Z. Yu, B. V. Kumar, and J. Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*, 2018. 2