MetaSAug: Meta Semantic Augmentation for Long-Tailed Visual Recognition

Shuang Li[1] Kaixiong Gong[1] Chi Harold Liu[1†] Yulin Wang[2] Feng Qiao[3] Xinjing Cheng[3]
[1]Beijing Institute of Technology [2]Tsinghua University [3]Inceptio Tech.
{shuangli, kxgong}@bit.edu.cn liuchi02@gmail.com wang-yl19@mails.tsinghua.edu.cn
feng.qiao@inceptio.ai cnorbot@gmail.com

Abstract

Real-world training data usually exhibits long-tailed distribution, where several majority classes have a significantly larger number of samples than the remaining minority classes. This imbalance degrades the performance of typical supervised learning algorithms designed for balanced training sets. In this paper, we address this issue by augmenting minority classes with a recently proposed implicit semantic data augmentation (ISDA) algorithm [37], which produces diversified augmented samples by translating deep features along many semantically meaningful directions. Importantly, given that ISDA estimates the class-conditional statistics to obtain semantic directions, we find it ineffective to do this on minority classes due to the insufficient training data. To this end, we propose a novel approach to learn transformed semantic directions with meta-learning automatically. In specific, the augmentation strategy during training is dynamically optimized, aiming to minimize the loss on a small balanced validation set, which is approximated via a meta update step. Extensive empirical results on CIFAR-LT-10/100, ImageNet-LT, and iNaturalist 2017/2018 validate the effectiveness of our method.

1. Introduction

Deep convolutional neural networks (CNNs) have achieved remarkable success in recent years [22, 15, 17]. Their state-of-the-art performance is typically demonstrated on the benchmarks such as ImageNet [31] and MS COCO [24]. While these datasets are established by ideally collecting a similar and sufficient number of samples for each class, real-world training data is usually imbalanced, as shown in Fig. 1(a). For example, in automatic medical diagnosis, a few common diseases may dominate the training set, with scarce cases for the remaining classes. This long-tailed distribution, unfortunately, degrades the performance of networks severely if using a standard training strategy (e.g., supervised learning with the cross-entropy loss).

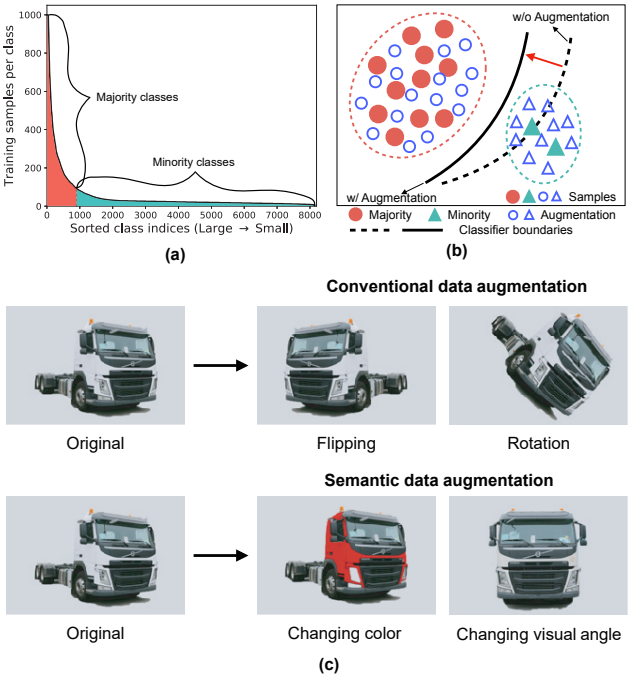

Figure 1. **(a)**: In the data distribution of the real-world dataset iNaturalist 2018, a few majority classes account for the most samples, while the minority classes are under-represented. **(b)**: Motivation of this work. Facilitating data augmentation for long-tailed problems to ameliorate the classifier performance. **(c)**: Illustration of traditional data augmentation and semantic data augmentation.

To address the issue of data imbalance, a natural solution might be to augment the minority classes for more training samples as shown in Fig. 1(b), e.g., by leveraging the data augmentation technique [15, 17, 42, 27]. However, conventional data augmentation techniques like cropping, mirroring and mixup are typically performed on the inputs. As a result, the diversity of augmented samples is inherently limited by the small amount of training data in minority classes.

Fortunately, this problem can potentially be solved by a recently proposed implicit semantic data augmentation (ISDA) technique [37]. ISDA performs class identity preserving semantic transformation (e.g., changing the color

†C. Liu is the corresponding author.

of an object and changing the visual angles) by translating deep features towards certain meaningful semantic directions as shown in Fig. 1(c). The deep feature space extracted by CNNs tends to be linearized and has significantly smaller complexity than the pixel space. Therefore, the minority classes will be effectively augmented for more diversity as long as proper semantic directions are found. ISDA estimates class-wise covariance matrices from deep features and sample semantic directions from a Gaussian distribution. Nevertheless, we find that this leads to inferior performance in the long-tailed scenario, since scarce data in minority classes is insufficient to obtain reasonable covariance matrices.

In this paper, we propose a meta semantic augmentation (MetaSAug) approach, aiming to perform effective semantic data augmentation for long-tailed problems via learning more meaningful class-wise covariance. Our major insight here is that *if the appropriate covariance matrices are used for semantic augmentation, the loss on a balanced validation set should be minimized*. At every training iteration, we perform validation on a small balanced validation set, and update the class-wise covariance by minimizing the validation loss. Specifically, we first fulfill the augmentation procedure using current class-wise covariance. Then, we calculate the loss on the validation set with respect to the class-wise covariance. By optimizing the validation loss, we can obtain the updated class-wise covariance that contains rich semantic directions. With it, we train the models on the augmentation set with sufficient semantically augmented samples. In addition, our method can be treated as a plug-in module and be unified with previous methods. We further improve the classification ability of focal loss [23] and LDAM loss [7] by combining them with MetaSAug.

We conduct extensive experiments on several long-tailed datasets, including the artificially long-tailed CIFAR-10/100 [21, 9], ImageNet [31, 26], and the naturally long-tailed dataset inaturalist 2017 and 2018 [36, 1]. The results demonstrate the effectiveness of our method.

2. Related Work

In this section, we briefly review the works related to ours.

Re-sampling. Researchers propose to achieve a more balanced data distribution by over-sampling the minority classes [32, 5, 6] or under-sampling the majority classes [14, 19, 5]. Although being effective, over-sampling might result in over-fitting of minority classes while under-sampling may weaken the feature learning of majority classes due to the absence of valuable instances [40, 7, 8, 9]. Chawla et al. [8] reveal that stronger augmentation for minority classes is beneficial to mitigate over-fitting, which complies with the goal of our method.

Re-weighting. Also termed as cost-sensitive learning, re-weighting aims to assign weights to training samples on either class or instance level. A classic scheme is to re-weight the classes with the weights that are inversely proportional to their frequencies [16, 38]. Cui et al. [9] further improve this scheme with proposed effective number. Recently, meta-class-weight [18] exploits meta-learning to estimate precise class-wise weights, while Cao et al. [7] allocate large margins to tail classes. Apart from above works, Focal Loss [23], L2RW [30] and meta-weight-net [33] assign weights to examples instance-wisely. Specifically, focal loss assigns weights according to the instance predictions, while L2RW and meta-weight-net allot weights based on the gradient directions. Instead of focusing on designing different weights for classes, our method mainly aims to augment the training set to overcome the imbalance issue.

In addition, for learning better representations, some approaches propose to seprate the training into two stages: representation learning and classifier re-balancing learning [7, 18, 10, 20]. BBN [43] further unifies the two stages to form a cumulative learning strategy.

Meta-learning and head-to-tail knowledge transfer. The recent development of meta-learning [11, 35, 3] inspires researchers to leverage meta-learning to handle class imbalance. A typical series of approaches is to learn the weights for samples with meta-learning [30, 18, 33]. Another pipeline of methods attempts to transfer the knowledge from head to tail classes. Wang et al. [38] adopt a meta learner to regress the network parameters. Liu et al. [26] exploit a memory bank to transfer the features. Yin et al. [41] and Liu et al. [25] propose to transfer intra-class variance from head to tail. Different from these works, our method attempt to automatically learn semantic directions for augmenting the minority classes, ameliorating the classifier performance.

Data augmentation is a canonical technique, widely adopted in CNNs for alleviating over-fitting. For example, rotation and horizontal flipping are employed for maintaining the prediction invariance of CNNs [15, 17, 34]. In complementary to the traditional data augmentation, semantic data augmentation that performs semantic altering is also effective for enhancing classifier performance [4, 37]. ISDA [37] performs semantic augmentation with the class-conditional statistics, but cannot estimate reasonable covariance with the scarce data in minority classes. The major difference between ours and ISDA is that MetaSAug utilizes meta-learning to learn proper class-wise covariance for achieving more meaningful augmentation results.

3. Method

Consider a training set $D = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$ with N training samples, where $\boldsymbol{x}_i$ denotes i-th training sample and y_i denotes its corresponding label over C class. Let f denote the classifier with parameter Θ and $\boldsymbol{a}_i$ denote the feature of

i-th sample extracted by classifier f. In the practical applications, the training set D is often imbalanced, resulting in poor performance on the minority classes. Therefore, we aim to perform semantic augmentation for minority classes, ameliorating the learning of classifiers.

3.1. Implicit Semantic Data Augmentation

Here, we revisit the implicit semantic data augmentation (ISDA) [37] approach. For semantic augmentation, ISDA statistically estimates the class-wise covariance matrices $\mathbf{\Sigma} = \{\mathbf{\Sigma}_1, \mathbf{\Sigma}_2, ..., \mathbf{\Sigma}_C\}$ from deep features at each iteration. Then, ISDA samples transformation directions from the Gaussian distribution $\mathcal{N}(0, \lambda\mathbf{\Sigma}_{y_i})$ to augment the deep feature $\boldsymbol{a}_i$, where λ is a hyperparameter for tuning the augmentation strength. Naturally, to explore all possible meaningful directions in $\mathcal{N}(0, \lambda\mathbf{\Sigma}_{y_i})$, one should sample a tremendous number of directions. Take a step further, if sampling infinite directions, ISDA derives the upper bound of the cross-entropy loss on all the augmented features:

$$\begin{aligned}\mathcal{L}_{ISDA} &= \sum_{i=1}^{N} L_\infty(f(\boldsymbol{x_i};\Theta), y_i; \mathbf{\Sigma}) \\ &= \sum_{i=1}^{N} -\log\left(\frac{e^{z_i^{y_i}}}{\sum_{c=1}^{C} e^{z_i^c + \frac{\lambda}{2}\Delta\boldsymbol{w}_{cy_i}^\top \mathbf{\Sigma}_{y_i} \Delta\boldsymbol{w}_{cy_i}}}\right),\end{aligned} \tag{1}$$

where z_i^c is the c-th element of the logits output of $\boldsymbol{x}_i$, $\Delta\boldsymbol{w}_{cy_i} = (\boldsymbol{w}_c - \boldsymbol{w}_{y_i})$ and $\boldsymbol{w}_c$ is the c-th column of the weight matrix of last fully connected layer. By optimizing this upper bound $\mathcal{L}_{ISDA}$, ISDA can fulfill the equivalent semantic augmentation procedure efficiently.

However, the performance of ISDA relies on the covairance matrices estimation. In the long-tailed scenario, we find that ISDA has unsatisfactory performance, since scarce data in minority classes is insufficient to achieve reasonable covariance matrices.

3.2. Meta Semantic Augmentation

To address the issue of class imbalance, we propose to augment the minority classes for more training samples. As aforementioned, the scarcity of data limits the effectiveness of semantic augmentation. Therefore, we attempt to learn appropriate class-wise covariance matrices for augmenting, leading to better performance on minority classes. The key idea is that *if the appropriate covariance matrices are used for semantic augmentation, the loss on a balanced validation set should be minimized*. In this work, we utilize meta-learning to achieve this goal.

The meta-learning objective. Generally, by leveraging L_∞ in eq. (1), we can train the classifier and simultaneously fulfill the semantic augmentation procedure. However, in the context of class-imbalanced learning, the majority classes dominate the training set. And from eq. (1), we can observe that the augmentation results depend on the training data. If we directly apply eq. (1), we in fact mainly augment the majority classes, which disobeys our goal.

Hence, to tackle this issue, we propose to unify the class-conditional weights in [9] with eq. (1) for down-weighting the losses of majority samples. The class-conditional weights are defined as $\epsilon_c \approx (1-\beta)/(1-\beta^{n_c})$, where n_c is the number of data in c-th class and β is the hype-parameter with a recommended value $(N-1)/N$. To sum up, the optimal parameters Θ^* can be calculated with the weighted loss on the training set:

$$\Theta^*(\mathbf{\Sigma}) = \arg\min_{\Theta} \sum_{i=1}^{N} \epsilon_i L_\infty(f(\boldsymbol{x_i};\Theta), y_i; \mathbf{\Sigma}) \tag{2}$$

If we treat covariance matrices $\mathbf{\Sigma}$ as training hyperparameters, we actually can search their optimal value on the validation set as [30, 33, 3, 28, 29]. Specifically, consider a small validation set $D^v = \{\boldsymbol{x}_i^v, y_i^v\}_{i=1}^{N^v}$, where N^v is the amount of total samples and $N^v \ll N$. The optimal class-wise covariance can be obtained by minimizing the following validation loss:

$$\mathbf{\Sigma}^* = \arg\min_{\mathbf{\Sigma}} \sum_{i=1}^{N^v} L_{ce}(f(\boldsymbol{x}_i^v; \Theta^*(\mathbf{\Sigma})), y_i^v), \tag{3}$$

where $L_{ce}(\cdot,\cdot)$ is the cross-entropy (CE) loss function. Since the validation set is balanced, we adopt vanilla CE loss to calculate the loss on validation set.

Online approximation. To obtain the optimal value of Θ and $\mathbf{\Sigma}$, we need to go through two nested loops, which can be cost-expensive. Hence, we exploit an online strategy to update Θ and $\mathbf{\Sigma}$ through one-step loops. Given current step t, we can obtain current covariance matrices $\mathbf{\Sigma}^t$ according to [37]. Next, we update the parameters of classifier Θ with following objective:

$$\widetilde{\Theta}^{t+1}(\mathbf{\Sigma}^t) \leftarrow \Theta^t - \alpha\nabla_\Theta \sum_{i=1}^{N} \epsilon_i L_\infty(f(\boldsymbol{x_i};\Theta^t), y_i; \mathbf{\Sigma}^t), \tag{4}$$

where α is the step size for Θ. After executing this step of backpropagation, we obtain the optimized parameters $\widetilde{\Theta}^{t+1}(\mathbf{\Sigma}^t)$. Then we can update the class-wise covariance $\mathbf{\Sigma}$ using the gradient produced by eq. (3):

$$\mathbf{\Sigma}^{t+1} \leftarrow \mathbf{\Sigma}^t - \gamma\nabla_{\mathbf{\Sigma}} \sum_{i=1}^{N^v} L_{ce}(f(\boldsymbol{x}_i^v; \widetilde{\Theta}^{t+1}(\mathbf{\Sigma}^t)), y_i^v), \tag{5}$$

where γ it the step size for $\mathbf{\Sigma}$. With the learned class-wise $\mathbf{\Sigma}$, we can ameliorate the parameters Θ of classifier as:

$$\Theta^{t+1} \leftarrow \Theta^t - \alpha\nabla_\Theta \sum_{i=1}^{N} \epsilon_i L_\infty(f(\boldsymbol{x_i};\Theta^t), y_i; \mathbf{\Sigma}^{t+1}). \tag{6}$$

Since the updated class-wise covariance matrices $\mathbf{\Sigma}^{t+1}$ are learned from balanced validation data, we could expect $\mathbf{\Sigma}^{t+1}$ help to learn better classifier parameters Θ^{t+1}. In practice, we adopt the generally used technique SGD to implement our algorithm. In addition, several previous works have demonstrated that training the networks without rebalancing strategy in the early stage learns better generalizable representations [20, 7, 18]. Hence, we first train classifiers with vanilla CE loss, then with MetaSAug. The training algorithm is shown in Algorithm. 1.

3.3. Discussion

In this work, we utilize meta-learning to learn proper covariance matrices for augmenting the minority classes. Hence, it's essential to find out what $\mathbf{\Sigma}$ have learned from the validation data. To investigate this question, we conduct singular value decomposition to extract the singular values for the covariance matrix $\mathbf{\Sigma}_r$ of the most rare class r:

$$\mathbf{\Sigma}_r = \mathbf{U}\mathbf{M}\mathbf{V}^\top, \tag{7}$$

where each element in the diagonal of $\mathbf{M}$ is the singular value of $\mathbf{\Sigma}_r$. Then, we illustrate the top-5 singular values (max-normalized) of $\mathbf{\Sigma}_r$ learned by ISDA and our MetaSAug. Principal component analysis demonstrates that the eigenvector with larger singular value will contain more information variations [39, 2]. From Fig. 2(a), we observe that the largest singular value of $\mathbf{\Sigma}_r$ learned by ISDA on imbalanced dataset is remarkably larger than other singular values, while the information signals of other eigenvectors with smaller singular values are suppressed. This sharp distribution of singular values implies that the $\mathbf{\Sigma}_r$ has less important principal components, which one may not be able to sample diversified transformation vectors with the $\mathbf{\Sigma}_r$. The reason is that ISDA can not estimate appropriate $\mathbf{\Sigma}_r$ with the scarce data of minority classes.

When MetaSAug applies the proposed meta-learning method to learn $\mathbf{\Sigma}_r$, the singular value distribution of $\mathbf{\Sigma}_r$ becomes relatively balanced, as shown in Fig. 2(b). Apart from the one with largest singular value, the other eigenvectors also contain great information variance. With the $\mathbf{\Sigma}_r$ that has a balanced singular value distribution, one may sample diverse transformation vectors, leading to better augmentation results. In summary, with our proposed meta-learning method, the learned covariance matrix $\mathbf{\Sigma}_r$ contains more important principal components, implying that it may contain more semantic directions. In addition, we further experimentally verify that our meta-learning method can help improve the performance of classifiers in Section 4.5.

4. Experiment

We evaluate our method on the following long-tailed datasets: CIFAR-LT-10, CIFAR-LT-100, ImageNet-LT,

Algorithm 1 Leaning algorithm of MetaSAug

Input: Training set D; validation set D^v; ending steps T_1 and T_2;

Output: Learned classifier parameter Θ

for t $\leq T_1$ **do**
  Sample a batch $B = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{|B|}$ from D
  Calculate loss $\mathcal{L}_B = \frac{1}{|B|}\sum_{i=1}^{|B|} L_{ce}(f(\boldsymbol{x}_i;\Theta), y_i)$
  Update $\Theta \leftarrow \Theta - \alpha\nabla_\Theta \mathcal{L}_B$
for $T_1 <$ t $\leq T_2$ **do**
  Sample a batch $B = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{|B|}$ from D
  Sample a batch $B^v = \{(\boldsymbol{x}_i^v, y_i^v)\}_{i=1}^{|B^v|}$ from D^v
  Obtain current covariance matrices $\mathbf{\Sigma}$
  Compute $\mathcal{L}_B = \sum_{i=1}^{|B|} \epsilon_i L_\infty(f(\boldsymbol{x}_i;\Theta), y_i; \mathbf{\Sigma})$
  Update $\widetilde{\Theta}(\mathbf{\Sigma}) \leftarrow \Theta - \alpha\nabla_\Theta \mathcal{L}_B$
  Compute $\mathcal{L}_{B^v} = \sum_{i=1}^{|B^v|} L_{ce}(f(\boldsymbol{x}_i^v;\widetilde{\Theta}(\mathbf{\Sigma})), y_i^v)$
  Update $\mathbf{\Sigma} \leftarrow \mathbf{\Sigma} - \gamma\nabla_{\mathbf{\Sigma}} \mathcal{L}_{B^v}$
  Calculate the loss with the updated $\mathbf{\Sigma}$
    $\widetilde{\mathcal{L}}_B = \sum_{i=1}^{|B|} \epsilon_i L_\infty(f(\boldsymbol{x}_i;\Theta), y_i; \mathbf{\Sigma})$
  update $\Theta \leftarrow \Theta - \alpha\nabla_\Theta \widetilde{\mathcal{L}}_B$

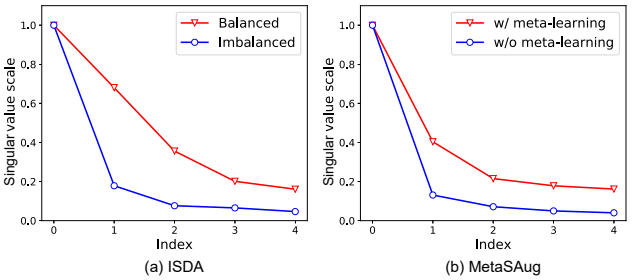

Figure 2. The top-5 singular values (max normalized) of covariance matrix $\mathbf{\Sigma}_r$ learned by ISDA and MetaSAug. **(a)**: "Balanced" refer to the covariance matrix estimated on balanced training set (CIFAR-10), while "Imbalanced" implies the $\mathbf{\Sigma}_r$ estimated on imbalanced training set (CIFAR-LT-10 with imbalance factor=200). **(b)**: Both experiments are conducted on the imbalanced set. The red and blue lines denote the $\mathbf{\Sigma}_r$ learned by MetaSAug with and without our meta-learning method, respectively.

iNaturalist 2017 and iNaturalist 2018. In addition, we report the average result of 3 random experiments. For those experiments conducted in the same settings, we directly quote their results from original papers. Code is available at https://github.com/BIT-DA/MetaSAug.

4.1. Datasets

Long-Tailed CIFAR is the long-tailed version of CIFAR dataset. The original CIFAR-10 (CIFAR-100) dataset consists of 50000 images drawn from 10 (100) classes with even data distribution. In other words, CIFAR-10 (CIFAR-100) has 5000 (500) images per class. Following [9], we discard some training samples to construct imbalanced datasets. We build 5 training sets by varying imbalance fac-

tor $\mu \in \{200, 100, 50, 20, 10\}$, where the μ denotes the image amount ratio between largest and smallest classes. If let n_i denotes the sample amount of i-th class, we can define $\mu = \frac{\max_i(n_i)}{\min_j(n_j)}$. As for test sets, we use the original balanced test sets. Following [18], we randomly select ten samples per class from training set to construct validation set D^v.

ImageNet-LT: ImageNet [31] is a classic visual recognition dataset, which contains 1,281,167 training images and 50,000 validation images. Liu et al. [26] build the long-tailed version of ImageNet, termed as ImageNet-LT. After discarding some training examples, ImageNet-LT remains 115,846 training examples in 1,000 classes. The imbalance factor is 1280/5. We adopt the original validation to test methods. In addition, Liu et al. [26] also construct a small balanced validation set with 20 images per class. Hence, we adopt ten images per class from it to construct our validation set D^v as [18].

iNaturalist 2017 and iNaturalist 2018. The iNaturalist datasets are large-scale datasets with images collected from real-world, which have an extremely imbalanced class distribution. The iNatualist 2017 [36] includes 579,184 training images in 5,089 classes with an imbalance factor of 3919/9, while the iNaturalist 2018 [1] is composed of 435,713 images from 8,142 classes with an imbalance factor of 1000/2. We adopt the original validation set to test our method. To construct the validation set D^v, we select five and two images from the training sets of iNaturalist 2017 and iNaturalist 2018, respectively, following [18].

4.2. Visual Recognition on CIFAR-LT

We conduct comparison experiments on the long-tailed datasets CIFAR-LT-10 and CIFAR-LT-100. Following [18, 9], we adopt the ResNet-32 [15] as the backbone network in our experiments.

Implementation details. For the baselines LDAM and LDAM-DRW, we reproduce them with the source code released by authors [7]. We train the ResNet-32 [15] with standard stochastic gradient decent (SGD) with momentum 0.9 and weight decay of 5×10^{-4} for all experiments. And We train the models on a single GPU for 200 epochs. In addition, we decay the learning rate by 0.01 at the 160^{th} and 180^{th} epochs as [18]. For our method, we adopt the initial learning rate 0.1. And we set the batch size as 100 for our experiments. The hyperparameter λ is selected from $\{0.25, 0.5, 0.75, 1.0\}$.

Comparison methods. We compare our method with the following methods:

- **Cross-entropy training** is the baseline method in long-tailed visual recognition, which trains ResNet-32 using vanilla cross-entropy loss function.

- **Class weighting methods**. This type of method assigns weights to training examples in class level, which includes class-balanced loss [9], meta-class-weight [18] and LDAM-DRW [7]. Class-balanced loss proposes effective number to measure the sample size of each class and the class-wise weights. Class-balanced focal loss and class-balanced cross-entropy loss refer to applying class-balanced loss on focal loss and cross-entropy loss, respectively. LDAM-DRW allocates label-aware margins to the examples based on the label distribution, and adopts deferred re-weighting strategy for better performance on tail classes.

- **Instance weighting methods** assign weights to samples according to the instance characteristic [30, 33, 23]. For example, focal loss [23] determine the weights for samples based on the sample difficulty. Though focal loss is not specially designed for long-tailed classification, it can penalize the samples of minority classes if the classifier overlooks the minority classes during training.

- **Meta-learning methods**. In fact, these methods adopt meta-learning to learn better class level or instance level weights [18, 33, 30]. For saving space, we only introduce them here. Meta-class-weight [18] exploits meta-learning to model the condition distribution difference between training and testing data, leading to better class level weights. While L2RW [30] and meta-weight-net [33] adopt meta-learning to model the instance-wise weights. L2RW directly optimizes the weight variables, while meta-weight-net additionally constructs a multilayer perceptron network to model the weighting function. Note that both L2RW and meta-weight-net can handle the learning with imbalanced label distribution and noisy labels.

- **Two-stage methods**. We also compared with methods that adopt two-stage learning [18, 7, 10]. BBN [43] unifies the representation and classifier learning stages to form a cumulative learning strategy.

Results. The experimental results of long-tailed CIFAR-10 with different imbalance factors are shown in Table 1, which are organized into three groups according to the adopted basic losses (i.e., cross-entropy, focal, and LDAM).

From the results, we can observe that re-weighting strategies are effective for the long-tailed problems, since several re-weighting methods (e.g., meta-class-weight) outperform the cross-entropy training by a large margin. We evaluate our method with the three basic losses. The results reveal that our method can consistently improve the performance of the basic losses significantly. Particularly, MetaSAug notably surpasses mixup that conducts augmentation on the inputs, manifesting that facilitating semantic augmentation is more effective in long-tailed scenarios. In addition, our method performs better than the re-weighting

Table 1. Test top-1 errors (%) of ResNet-32 on CIFAR-LT-10 under different imbalance settings. * indicates results reported in original paper. † indicates results reported in [18].

Imbalance factor	200	100	50	20	10
Cross-entropy training	34.13	29.86	25.06	17.56	13.82
Class-balanced cross-entropy loss [9]	31.23	27.32	21.87	15.44	13.10
Class-balanced fine-tuning† [10]	33.76	28.66	22.56	16.78	16.83
BBN* [43]	-	20.18	17.82	-	11.68
Mixup [42] (results from [43])	-	26.94	22.18	-	12.90
L2RW† [30]	33.75	27.77	23.55	18.65	17.88
Meta-weight net† [33]	32.80	26.43	20.90	15.55	12.45
Meta-class-weight with cross-entropy loss† [18]	29.34	23.59	19.49	13.54	11.15
MetaSAug with cross-entropy loss	**23.11**	**19.46**	**15.97**	**12.36**	**10.56**
Focal loss† [23]	34.71	29.62	23.29	17.24	13.34
Class-balanced focal loss† [9]	31.85	25.43	20.78	16.22	12.52
Meta-class-weight with focal loss† [18]	25.57	21.10	17.12	13.90	11.63
MetaSAug with focal loss	**22.73**	**19.36**	**15.96**	**12.84**	**10.74**
LDAM loss[7]	33.25	26.45	21.17	16.11	12.68
LDAM-DRW [7]	25.26	21.88	18.73	15.10	11.63
Meta-class-weight with LDAM loss † [18]	22.77	20.00	17.77	15.63	12.60
MetaSAug with LDAM loss	**22.65**	**19.34**	**15.66**	**11.90**	**10.32**

Table 2. Test top-1 errors (%) of ResNet-32 on CIFAR-LT-100 under different imbalance settings. * indicates results reported in original paper. † indicates results reported in [18].

Imbalance factor	200	100	50	20	10
Cross-entropy training	65.30	61.54	55.98	48.94	44.27
Class-balanced cross-entropy loss [9]	64.44	61.23	55.21	48.06	42.43
Class-balanced fine-tuning† [10]	61.34	58.50	53.78	47.70	42.43
BBN* [43]	-	57.44	52.98	-	40.88
Mixup [42] (results from [43])	-	60.46	55.01	-	41.98
L2RW† [30]	67.00	61.10	56.83	49.25	47.88
Meta-weight net† [33]	63.38	58.39	54.34	46.96	41.09
Meta-class-weight with cross-entropy loss† [18]	60.69	56.65	51.47	44.38	40.42
MetaSAug with cross-entropy loss	**60.06**	**53.13**	**48.10**	**42.15**	**38.27**
Focal loss† [23]	64.38	61.59	55.68	48.05	44.22
Class-balanced focal loss† [9]	63.77	60.40	54.79	47.41	42.01
Meta-class-weight with focal loss† [18]	60.66	55.30	49.92	44.27	40.41
MetaSAug with focal loss	**59.78**	**54.11**	**48.38**	**42.41**	**38.94**
LDAM loss [7]	63.47	59.40	53.84	48.41	42.71
LDAM-DRW [7]	61.55	57.11	52.03	47.01	41.22
Meta-class-weight with LDAM loss† [18]	60.47	55.92	50.84	47.62	42.00
MetaSAug with LDAM loss	**56.91**	**51.99**	**47.73**	**42.47**	**38.72**

methods. This demonstrates that our augmentation strategy indeed can ameliorate the performance of classifiers. When the dataset is less imbalanced (implying imbalance factor=10), our method can still stably achieve performance gains, revealing that MetaSAug won't damage the performance of classifier under the relatively balanced setting.

Table 2 presents the classification error of dataset long-tailed CIFAR-100, from which we can still observe that our methods achieve the best results in each group. Particularly, "MetaSAug with LDAM loss" exceeds the best competing method "Meta-class-weight with LDAM loss" by 3.56%.

4.3. Visual Recognition on ImageNet-LT

We use ResNet-50 [15] as the backbone network in the experiments on ImageNet-LT. And we train ResNet-50 with batch size 64. We decay the learning rate by 0.1 at 60^{th} epoch and 80^{th} epoch. In addition, for training efficiency, we only finetune the last full-connected layer while fixing

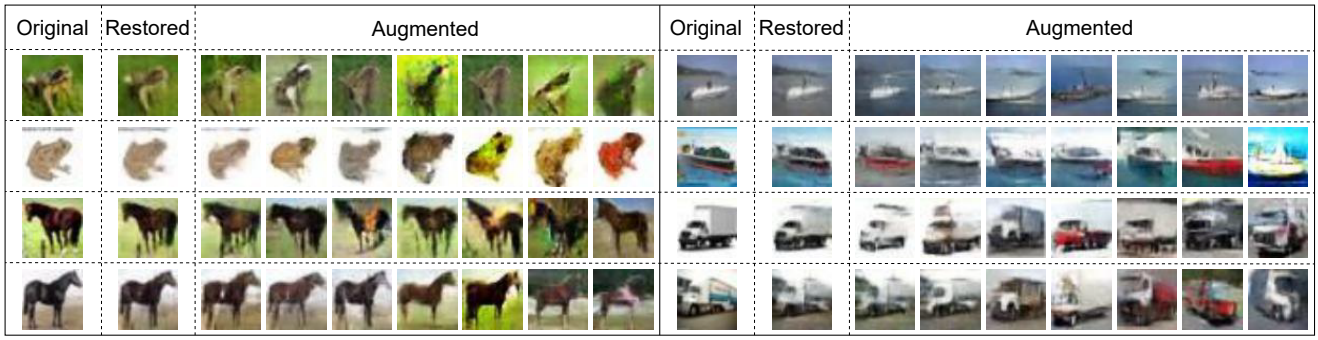

Figure 3. Visualization of the augmented examples for the four rarest classes: frog, horse, ship and truck (frequent → rare). We adopt WGAN-GP [13] generator to search the images corresponding to the augmented features. "Original" refers to the original training samples. "Restored" and "Augmented" present the original and augmented images generated by the generator, respectively. Our method is able to semantically alter the semantic of training images, e.g., changing color of objects, backgrounds, shapes of objects, etc.

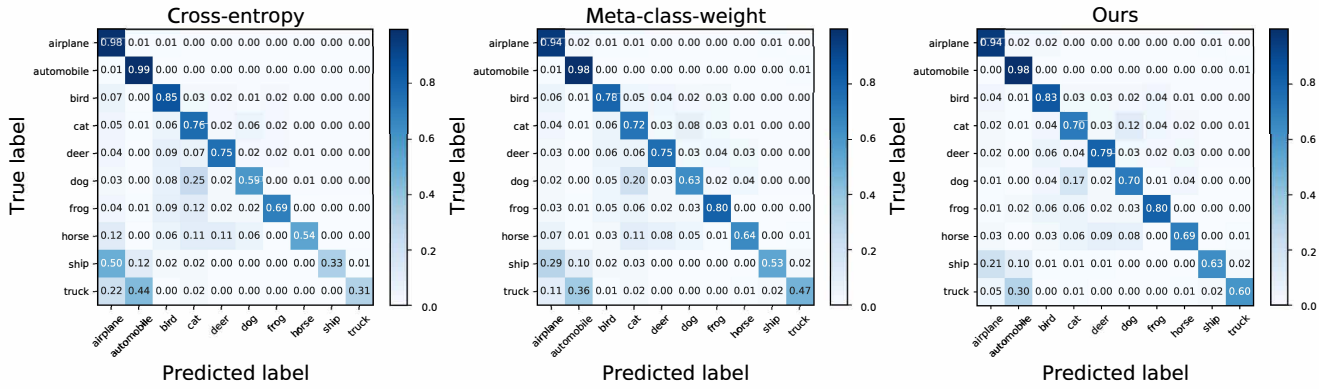

Figure 4. Illustration of confusion matrices of the vanilla cross-entropy training, meta-class-weight [18], and our method on dataset CIFAR-LT-10. The imbalance factor is 200. Classes are ranked by the frequency, i.e., frequent (left) → rare (right).

the representations in the meta-learning stage. We reproduce the comparison methods based on the code released by authors.

Results. The experimental results are shown in Table 3. Class-balanced cross-entropy performs better than cross-entropy training and LDAM-DRW surpasses LDAM by a large margin. These results imply that re-weighting strategy is also effective for the dataset with a large number of classes. Hence, MetaSAug also adopts this strategy to better fulfill the semantic augmentation procedure. In addition, compared with the best competing method meta-class-weight, MetaSAug can still yield better results, demonstrating that MetaSAug is able to perform data augmentation useful for the classification learning of classifiers.

4.4. Visual Recognition on iNaturalist Datasets

For fair comparisons, we adopt ResNet-50 [15] as the backbone network for iNaturalist 2017 and 2018. Following [18], we pre-train the backbone network on ImageNet for iNaturalist 2017. As for iNaturalist 2018, the network is pre-trained on ImageNet and iNaturalist 2017. We use stochastic gradient descent (SGD) with momentum to train models. The batch size is set as 64 and the initial learning rate is 0.01. In the meta-learning stage of our method, we decay the learning rate to 0.0001 and only finetune the last fully-connected layer for training efficiency.

Table 3. Test top-1 error rate (%) on ImageNet-LT of different models. (CE=Cross-entropy)

Method	Top-1 error
CE training	61.12
Class-balanced CE [9]	59.15
OLTR [26]	59.64
LDAM [7]	58.14
LDAM-DRW [7]	54.26
Meta-class-weight with CE loss [18]	55.08
MetaSAug with CE loss	**52.61**

Results. Table 4 presents the experimental results on the naturally-skewed datasets iNaturalist 2017 and iNaturalist 2018. MetaSAug and meta-class-weight both exploit the CE loss as basic loss. Compared with the improvement brought by class-balanced CE [9] to CE Loss, MetaSAug

Table 4. Test top-1 error rate (%) on iNaturalist (iNat) 2017 and 2018 of different models. *results is quoted from original papers. † indicates results reported in [18]. (CE=Cross-entropy)

Method	iNat 2017	iNat 2018
CE	43.21	34.24
Class-balanced CE [9]	42.02	33.57
Class-balanced focal* [9]	41.92	38.88
cRT* [20]	-	32.40
BBN* [43]	**36.61**	33.71
LDAM* [7]	-	35.42
LDAM [7]	39.15	34.13
LDAM-DRW* [7]	-	32.00
LDAM-DRW [7]	37.84	32.12
Meta-class-weight† [18]	40.62	32.45
MetaSAug	36.72	**31.25**

further enhances the performance of CE loss, implying that performing effective data augmentation is also of importance for long-tailed classification. In addition, MetaSAug yields the best results among these competitive methods on iNat 2018 and is on par with the state-of-art method BBN [43] on iNat 2017. These results demonstrate that our method indeed can facilitate data augmentation useful for classification in the deep feature space.

4.5. Analysis

Ablation study. To verify each component of MetaSAug, we conduct ablation study (see Table 5). Removing re-weighting or meta-learning causes performance drop. This manifests 1) re-weighting is effective to construct a proper meta-learning objective, and 2) our meta-learning method can indeed learn covariance useful for classification. Importantly, the latter is non-trivial since MetaSAug achieves the notable accuracy gains as shown in Table 5. While this cannot be reached by meta-weighting methods with fixed ISDA (e.g., L2RW; Meta-weight net, MWN; Meta-class-weight, MCW). Furthermore, we observe that ISDA can boost former long-tailed methods to some extent, but the improvement is limited. This also validate the importance of our meta-learning algorithm.

Adaptivity to deeper backbone networks. For a reasonable comparison with baselines, we adopt the commonly used ResNet-32 and ResNet-50 to evaluate our method on CIFAR-LT and ImageNet-LT/iNaturalist datasets, respectively. However, MetaSAug can be easily adapted to other networks, and, as indicated in [37], deeper models may even benefit our method more due to their stronger ability to model complex semantic relationships. In Table 6, we show the results of MetaSAug, MCW [18] and LDAM-DRW [7] with different backbone networks. One can observe that MetaSAug consistently outperforms other methods.

Confusion matrices. To find out whether our method ameliorates the performance on minority classes, we plot the confusion matrices of cross-entropy (CE) training, meta-class-weight [18], and our method on CIFAR-LT-10 with imbalance factor 200. From Fig.4, we can observe that CE training can almost perfectly classify the samples in majority classes, but suffers severe performance degeneration on the minority classes. Due to the proposed two-component weighting, meta-classes-weight performs much better than CE training on the minority classes. Since MetaSAug inclines to augment the minority classes, it can further enhance the performance on rare classes and reduce the confusion between similar classes (implying automobile & truck, and airplane & ship).

Table 5. Ablation study of MetaSAug using cross-entropy loss on dataset CIFAR-LT-10. The results are top-1 errors (%).

Imbalance factor	100	50	20
MetaSAug w/o re-weighting	25.96	20.63	15.15
MetaSAug w/o meta-learning	21.68	17.43	13.08
MetaSAug	**19.46**	**15.97**	**12.36**
ISDA, L2RW [30]	25.16	20.78	16.53
ISDA, MWN [33]	24.69	20.42	14.71
ISDA, MCW [18]	20.78	17.12	12.93

Visualization Results. To intuitively reveal that our method can indeed alter the semantics of training examples and generate diverse meaningful augmented samples, we carry out the visualization experiment (the detailed visualization algorithm is presented in [37]). The visualization results are shown in Fig. 3, from which we can observe that MetaSAug is capable of semantically altering the semantics of training examples while preserving the label identity. Particularly, MetaSAug can still generate meaningful augmented samples for the rarest class "truck".

Table 6. Test top-1 errors (%) on ImageNet-LT of methods with different backbone networks.

Network	MCW [18]	LDAM-DRW [7]	MetaSAug
ResNet-50	55.08	54.26	**52.61**
ResNet-101	53.76	53.55	**50.95**
ResNet-152	53.18	52.86	**49.97**

5. Conclusion

In this paper, we delve into the long-tailed visual recognition problem and propose to tackle it from a data augmentation perspective, which has not been fully explored yet. We present a meta semantic augmentation (MetaSAug) approach that learn appropriate class-wise covariance matrices for augmenting the minority classes, ameliorating the learning of classifiers. In addition, MetaSAug is orthogonal to several former long-tailed methods, e.g., LDAM and focal loss. Extensive experiments on several benchmarks validate the effectiveness and versatility of MetaSAug.

References

[1] iNaturalist 2018 competition dataset.https://github.com/visipedia/inat_comp, 2018. 2, 5

[2] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010. 4

[3] Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. Learning to learn by gradient descent by gradient descent. In *NeurIPS*, pages 3981–3989, 2016. 2, 3

[4] Antreas Antoniou, Amos Storkey, and Harrison Edwards. Data augmentation generative adversarial networks. *arXiv preprint arXiv:1711.04340*, 2017. 2

[5] Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, 2018. 2

[6] Jonathon Byrd and Zachary Lipton. What is the effect of importance weighting in deep learning? In *ICML*, pages 872–881, 2019. 2

[7] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In *NeurIPS*, pages 1567–1578, 2019. 2, 4, 5, 6, 7, 8

[8] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002. 2

[9] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *CVPR*, pages 9268–9277, 2019. 2, 3, 4, 5, 6, 7, 8

[10] Yin Cui, Yang Song, Chen Sun, Andrew Howard, and Serge Belongie. Large scale fine-grained categorization and domain-specific transfer learning. In *CVPR*, pages 4109–4118, 2018. 2, 5, 6

[11] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 2017. 2

[12] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

[13] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans. In *NeurIPS*, pages 5769–5779, 2017. 7

[14] Haibo He and Edwardo A Garcia. Learning from imbalanced data. *TKDE*, 21(9):1263–1284, 2009. 2

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1, 2, 5, 6, 7

[16] Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. Learning deep representation for imbalanced classification. In *CVPR*, pages 5375–5384, 2016. 2

[17] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, pages 4700–4708, 2017. 1, 2

[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 *CVPR*, pages 7610–7619, 2020. 2, 4, 5, 6, 7, 8

[19] Nathalie Japkowicz and Shaju Stephen. The class imbalance problem: A systematic study. *Intelligent data analysis*, 6(5):429–449, 2002. 2

[20] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. *arXiv preprint arXiv:1910.09217*, 2019. 2, 4, 8

[21] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 2

[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017. 1

[23] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pages 2980–2988, 2017. 2, 5, 6

[24] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. Springer, 2014. 1

[25] Jialun Liu, Yifan Sun, Chuchu Han, Zhaopeng Dou, and Wenhui Li. Deep representation learning on long-tailed data: A learnable embedding augmentation perspective. In *CVPR*, pages 2970–2979, 2020. 2

[26] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *CVPR*, pages 2537–2546, 2019. 2, 5, 7

[27] Alexander J Ratner, Henry Ehrenberg, Zeshan Hussain, Jared Dunnmon, and Christopher Ré. Learning to compose domain-specific transformations for data augmentation. In *NeurIPS*, pages 3236–3246, 2017. 1

[28] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2017. 3

[29] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. Meta-learning for semi-supervised few-shot classification. *arXiv preprint arXiv:1803.00676*, 2018. 3

[30] Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. *arXiv preprint arXiv:1803.09050*, 2018. 2, 3, 5, 6, 8

[31] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, and Michael Bernstein. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015. 1, 2, 5

[32] Li Shen, Zhouchen Lin, and Qingming Huang. Relay backpropagation for effective learning of deep convolutional neural networks. In *ECCV*, pages 467–482. Springer, 2016. 2

[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 *NeurIPS*, pages 1919–1930, 2019. 2, 3, 5, 6, 8

[34] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2

[35] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, pages 4077–4087, 2017. 2

[36] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The iNaturalist species classification and detection dataset. In *CVPR*, 2018. 2, 5

[37] Yulin Wang, Xuran Pan, Shiji Song, Hong Zhang, Gao Huang, and Cheng Wu. Implicit semantic data augmentation for deep networks. In *NeurIPS*, pages 12635–12644, 2019. 1, 2, 3, 8

[38] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Learning to model the tail. In *NeurIPS*, pages 7029–7039, 2017. 2

[39] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987. 4

[40] Tong Wu, Qingqiu Huang, Ziwei Liu, Yu Wang, and Dahua Lin. Distribution-balanced loss for multi-label classification in long-tailed datasets. *arXiv preprint arXiv:2007.09654*, 2020. 2

[41] Xi Yin, Xiang Yu, Kihyuk Sohn, Xiaoming Liu, and Manmohan Chandraker. Feature transfer learning for face recognition with under-represented data. In *CVPR*, pages 5704–5713, 2019. 2

[42] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018. 1, 6

[43] Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen. Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *CVPR*, pages 9719–9728, 2020. 2, 5, 6, 8