– Supplemental Material –
Distilling Virtual Examples for Long-tailed Recognition

Yin-Yin He[1], Jianxin Wu[1], Xiu-Shen Wei[2,1]

[1]State Key Laboratory for Novel Software Technology, Nanjing University, China

[2]School of Computer Science and Engineering, Nanjing University of Science and Technology, China

heyy@lamda.nju.edu.cn, {wujx2001, weixs.gm}@gmail.com

A. Additional experiments on the influence of virtual example distribution flatness

Due to the space limit of the main paper, we explain the experimental details of Sec 3.3 in this appendix. The **virtual example ratio** R between tail and head equals $\frac{n_{tail}+n_{head}\times\epsilon}{n_{head}-n_{head}\times\epsilon}$ after smoothing, and $R \in [\frac{n_{tail}}{n_{head}}, \frac{2n_{tail}}{n_{head}}+1)$ because we restrict that $0 \leq \epsilon < 0.5$. We sampled $\epsilon \in \{0, 0.1, 0.2, 0.3, 0.4\}$, and each experiment was run for 5 times to compute the mean and standard deviation.

In this section, following Sec. 3.3 of our main paper, we conduct additional experiments under different settings, to further justify our conjecture that virtual example distribution must be flat. Specifically, we vary the categories and the imbalance factor. Results are in Fig. 1.

In Fig. 1(a), we use "airplane" as the head and "dog" as the tail category, which are very dissimilar in appearance. But, similar to what Fig. 3 in the main paper shows, the performance is improved significantly as the virtual example distribution gets flatter. Comparing Fig. 1(a), Fig. 1(b) and Fig. 1(c), under different imbalance factors or dataset size, all head accuracies are almost intact while the tail accuracies increase significantly as the virtual example distribution goes flatter.

All these observations are consistent with our conjecture that the virtual example distribution must be flat.

B. Implementation details

In this section, we describe the implementation details of our DiVE in different long-tailed datasets. The properties of all datasets used in our experiments are summarized in Table 1.

On CIFAR-100-LT. CIFAR-100 contains 100 categories and 60,000 images (50,000 for training and 10,000 for validation). Following [9], we manually split the long-tailed versions of it with controllable degrees of data imbalance.

Table 1. Properties of long-tailed datasets. For CIFAR-100-LT, we report results with different imbalance factors.

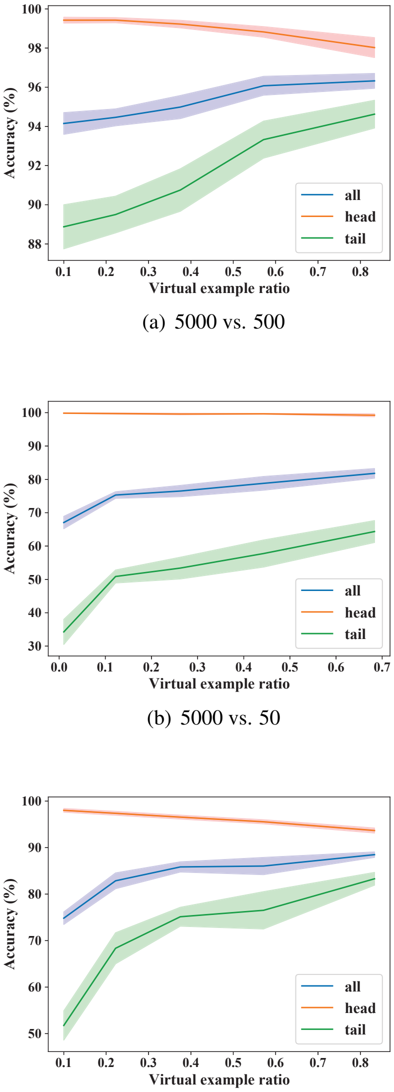
Dataset	#Classes	Imbalance Factor
CIFAR-100-LT	100	10, 50, 100
ImageNet-LT	1,000	256
iNaturalist2018	8,142	500

We follow the data augmentation strategy in [3]: randomly crop a 32 × 32 patch from the original image or its horizontal flip with 4 pixels padded on each side. ResNet-32 [3] is used as the backbone network. Following [9], we use stochastic gradient descent (SGD) to optimize networks with momentum of 0.9, weight decay of 2×10^{-4} for 200 epochs with batch size being 128. The initial learning rate is 0.1 with first 5 epochs being linear warm-up, then decayed at 120^{th} and 160^{th} epochs by 0.01. In the proposed DiVE method, we choose $\tau = 3$ with the power normalization ($p = 0.5$), as well as $\alpha = 0.5$ in all experiments on CIFAR-100-LT.

On ImageNet-LT. It is a long-tailed version of ImageNet, first used by [6]. It has 115.8K images from 1000 categories, with $n_{\text{max}} = 1280$ and $n_{\text{min}} = 5$.

To have fair comparisons, we use ResNeXt-50 [8] as the backbone network in all experiments on ImageNet-LT. We use the same data augmentation strategy as that in [6] and [5]. In detail, images are firstly resized by setting shorter side to 256, then we randomly take a 224 × 224 crop from it or its horizontal flip, followed by color jittering. For training strategies, we follow [5]. Both teacher and student networks are trained for 90 epochs with batch size 512. The initial learning rate is set to 0.2 and cosine decayed epoch by epoch. Mini-batch stochastic gradient descent (SGD) with momentum of 0.9, weight decay of 5×10^{-4} is used as our optimizer. In this dataset, power normalization is not chosen in DiVE, and we set $\tau = 9$, $\alpha = 0.5$.

On iNaturalist. The iNaturalist species classification datasets are large-scale real-world datasets with severe long-tail problems. iNaturalist2018 [1] contains 437.5K images from 8,412 categories, with $\beta = 500$. We adopt the official training and validation split in our experiments.



(a) 5000 vs. 500

(b) 5000 vs. 50

(c) 500 vs. 50

Figure 1. Accuracy (mean value and plus / minus 1 standard deviation) in “airplane” vs. “dog” binary classification experiments. We take “airplane” as the head and “dog” as the tail categories. The numbers in each sub-figure title are the number of samples in the head and tail, respectively.

We use ResNet-50 [3] as the backbone network across all experiments for iNaturalist2018. Standard data augmentation strategies proposed in [2] are utilized. We train the teacher and student networks both for 90 epochs with batch size 256. The initial learning rate is set to 0.1, and decayed following the cosine decay schedule. The optimizer is the same as that used for ImageNet-LT. In DiVE, we set $\tau = 2$ with the power normalization ($p = 0.5$) and $\alpha = 0.5$. Some methods reported results trained with 200 epochs, hence we also report DiVE results with 200 epochs.

Note that the training strategies of RIDE [7] are slightly different from those standard long-tailed training strategies. So, when comparing with RIDE [7], we follow experimental settings in [7]. For implementation details of RIDE-DiVE, we adopt BSCE to train a 6 experts RIDE in place of LDAM. Then we distill the virtual examples to each expert of a 4 experts student network using Eqn. (15) in our main paper, and train the expert assignment module finally. We normalize the feature and classifier weights of student network for fair comparison.

In addition, we set α to 0.75 in all experiments, and set $\tau = 3$ in ImageNet-LT for RIDE-DiVE because the teacher networks provide more reliable predictions.

C. Results on various shifted test label distributions

Recently, [4] proposed a more realistic evaluation protocol, they evaluated models on a range of target label distributions, including two types, Forward and Backward. For the Forward type, the target label distribution becomes similar to the source label distribution when the imbalance factor increases. The order is flipped for the Backward type. Please refer to [4] for more details.

Follow [4], we evaluate CE, BSCE and DiVE trained for 90 epochs on test time shifted ImageNet-LT, the results are in Table 2. Here PC means injecting target label distribution information to the final output. Knowing the target label distribution or not, DiVE surpass CE and BSCE by a large margin.

D. t-SNE visualization

We use the t-SNE method to visualize the embedding space on CIFAR100-LT ($\beta = 100$). We aggregate the classes into ten groups, based on the order of the number of examples from head to tail, and sample one class from each group for visualization. Results are in Fig. 2. In CE (cross entropy), the feature embedding is dispersed for both head and tail, making it hard to distinguish classes of similar appearance (e.g., “mouse” and “squirrel”). DiVE enlarges the inter-class variance while reduces the intra-class variance for both head and tail (e.g., features of “mouse” and “squirrel” are more compact and easier to separate). And, RIDE-DiVE is better than both.

E. Sample images visualization

In Fig. 3, we visualize some sample images in ImageNet-LT test set, comparing the predictions of CE, BSCE and DiVE. We choose samples on which DiVE’s predictions are correct, to show how dose DiVE correct the predictions.

DiVE can correct the predictions not only to semantically “nearby” categories (e.g., the “Polyporus frondosus”

Table 2. Top-1 accuracy over all classes on test time shifted ImageNet-LT. All models are trained for 90 epochs.

Dataset	Forward					Uniform	Backward				
Imbalance factor	50	25	10	5	2	1	2	5	10	25	50
CE	61.67	59.48	56.01	52.84	48.07	43.89	39.66	34.35	30.70	26.54	23.95
BSCE	59.46	58.51	56.64	54.94	52.50	50.48	48.24	5.29	43.18	40.89	39.31
DiVE	**62.61**	**61.44**	**59.73**	**58.06**	**55.40**	**53.10**	**50.88**	**47.87**	**45.69**	**43.17**	**41.55**
PC CE	61.91	59.80	56.60	54.39	51.39	49.33	47.71	46.20	45.57	45.03	45.41
PC BSCE	63.31	61.32	58.16	55.72	52.55	50.48	48.73	47.48	46.81	46.74	47.09
PC DiVE	**65.82**	**63.56**	**60.70**	**58.38**	**55.17**	**53.10**	**51.39**	**49.97**	**49.42**	**49.15**	**49.29**

example and the "Siberian husky" example in Fig. 3), but also to semantically "far" categories (e.g., the "pot" example and the "ski mask" example in Fig. 3).

References

[1] Yin Cui, Yang Song, Chen Sun, Andrew Howard, and Serge Belongie. Large scale fine-grained categorization and domain-specific transfer learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4109–4118, 2018. 1

[2] 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. 2

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 770–778, 2016. 1, 2

[4] Youngkyu Hong, Seungju Han, Kwanghee Choi, Seokjun Seo, Beomsu Kim, and Buru Chang. Disentangling label distribution for long-tailed visual recognition. *arXiv preprint arXiv:2012.00321*, 2020. 2

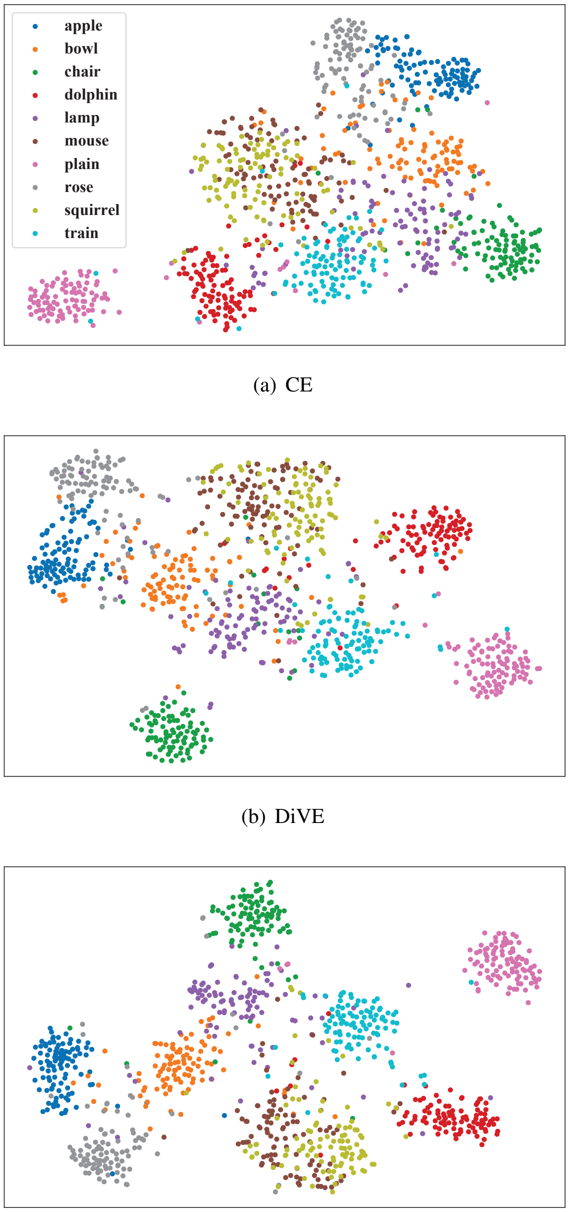
[5] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *Int. Conf. Learn. Represent.*, 2020. 1

[6] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X. Yu. Large-scale long-tailed recognition in an open world. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 1

[7] Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella X Yu. Long-tailed recognition by routing diverse distribution-aware experts. In *Int. Conf. Learn. Represent.*, 2021. 2

[8] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1492–1500, 2017. 1

[9] Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen. BBN: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9719–9728, 2020. 1



(a) CE

(b) DiVE

(c) RIDE-DiVE

Figure 2. t-SNE visualization of different models' embedding space on CIFAR100-LT ($\beta = 100$).

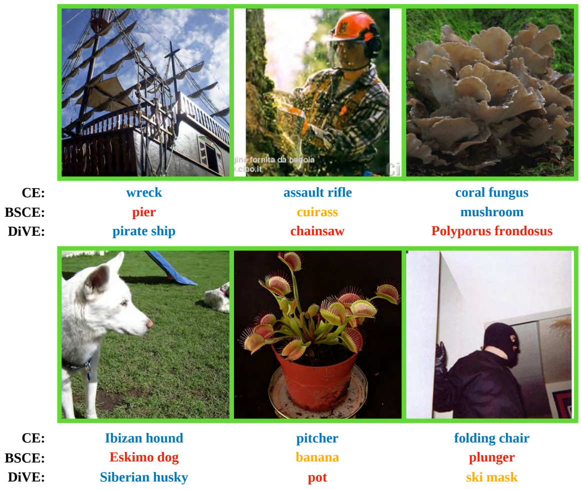


Figure 3. Some sample images in ImageNet-LT test set with predictions from CE, BSCE and DiVE. Below each image are the predicted categories from CE, BSCE and DiVE on it. Categories in blue are "Many", categories in yellow are "Medium", while categories in red are "Few". DiVE's predictions are also ground-truth labels.