南京航空航天大學

Nanjing University of Aeronautics and Astronautics

ParNeC 模式识别与神经计算研究组

PAttern Recognition and NEural Computing

Tackling the long-tailed problem from the perspective of knowledge distillation

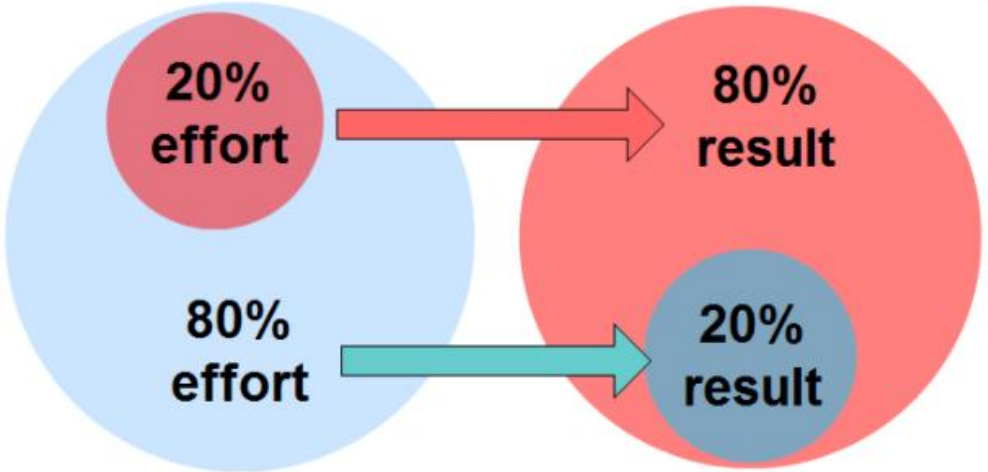
- Self Supervision to Distillation for Long-Tailed Visual Recognition
- Distilling Virtual Examples for Long-tailed Recognition

2021.11.01

Long-Tailed Distribution

What is the long-tailed distribution?

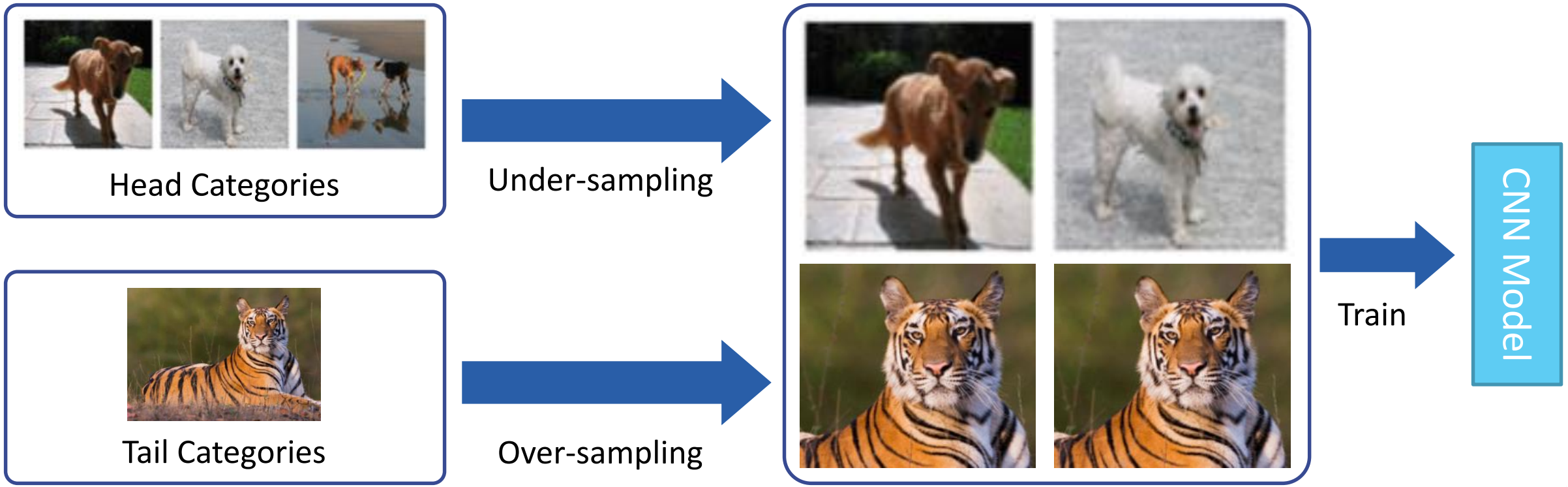
- NLP → Zipf's Law
- Economics → Pareto Principle
- Computer Vision



Re-balancing(Re-Sampling/Re-Weighting)

- Re-Sampling: class-balanced sampling
  - Over-sampling for the tail categories.
  - Under-sampling for the head categories.

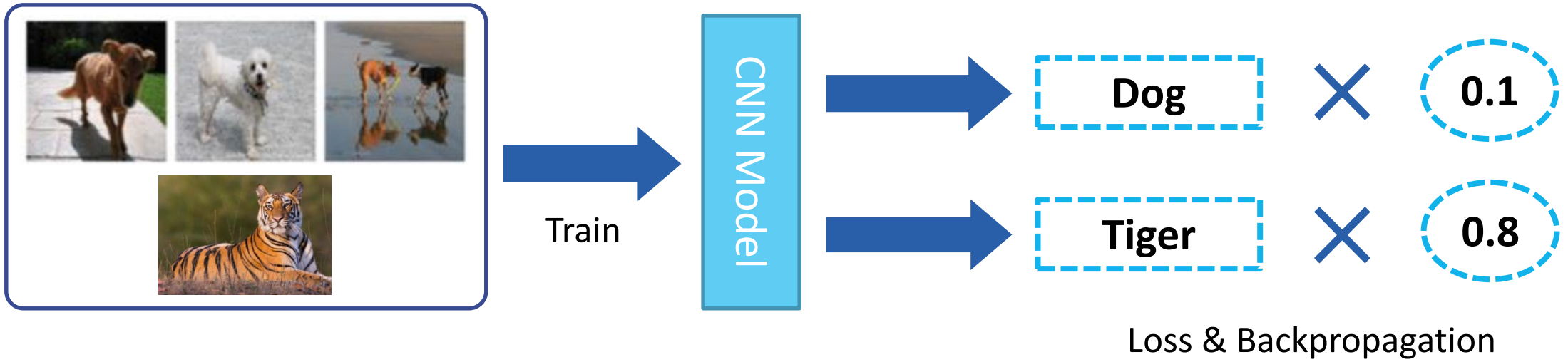
- Drawbacks
  - Over-fitting to the tail.
  - Under-fitting to the head.



Re-balancing(Re-Sampling/Re-Weighting)

- Re-Weighting
  - Weighting by inverse class frequency.
  - Weighting by inverse square root of class frequency.

- Drawbacks
  - Over-fitting to the tail.
  - Under-fitting to the head.



BBN

Bilateral-Branch Network with Cumulative Learning for Long-Tailed Visual Recognition .CVPR2020

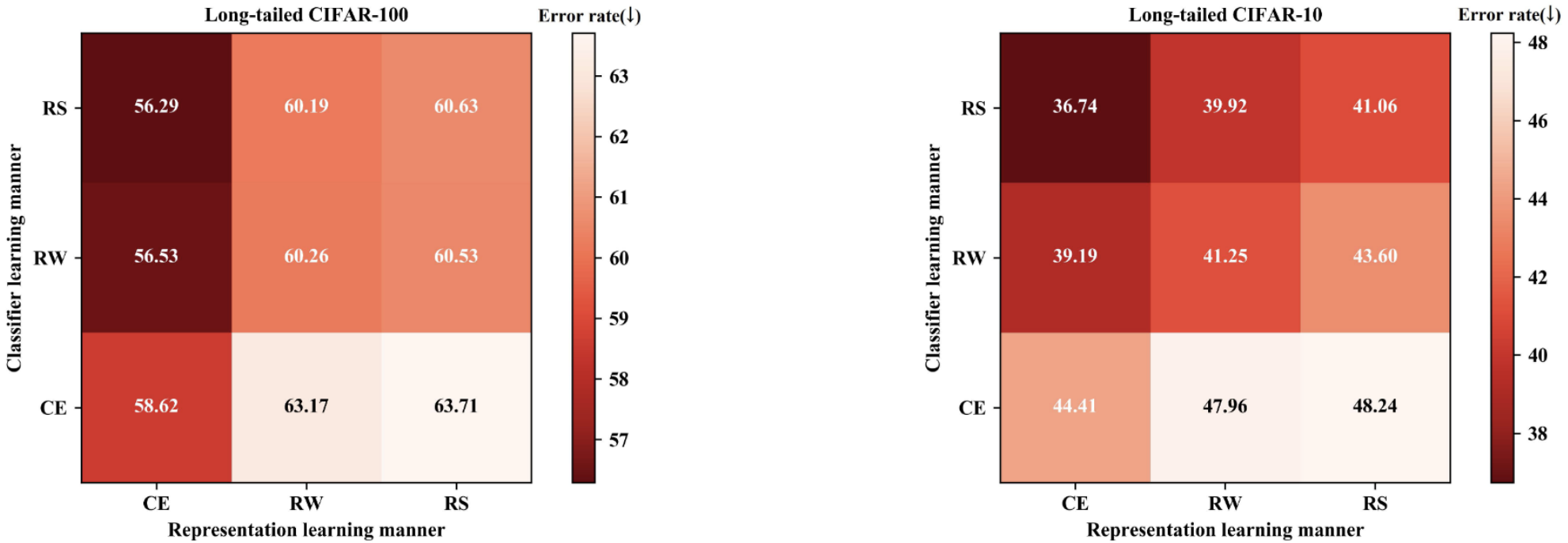


Figure 2. Top-1 error rates of different manners for representation learning and classifier learning on two long-tailed datasets CIFAR-100-IR50 and CIFAR-10-IR50 [3]. "CE" (Cross-Entropy), "RW" (Re-Weighting) and "RS" (Re-Sampling) are the conducted learning manners. As observed, when fixing the representation (comparing error rates of three blocks in the vertical direction), the error rates of classifiers trained with RW/RS are reasonably lower than CE. While, when fixing the classifier (comparing error rates in the horizontal direction), the representations trained with CE surprisingly get lower error rates than those with RW/RS. Experimental details can be found in Section 3.

SSL (rotation)

UNSUPERVISED REPRESENTATION LEARNING BY PREDICTING IMAGE ROTATIONS.ICLR 2018

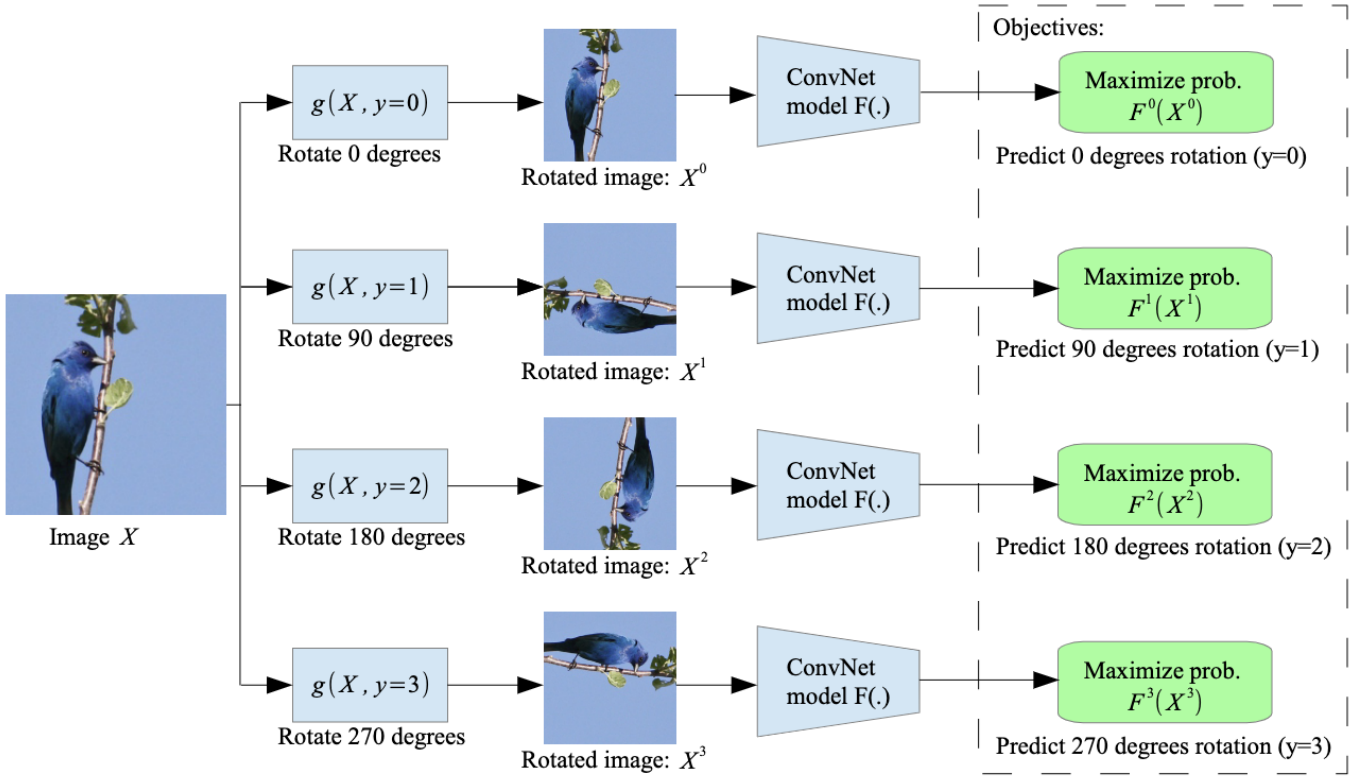
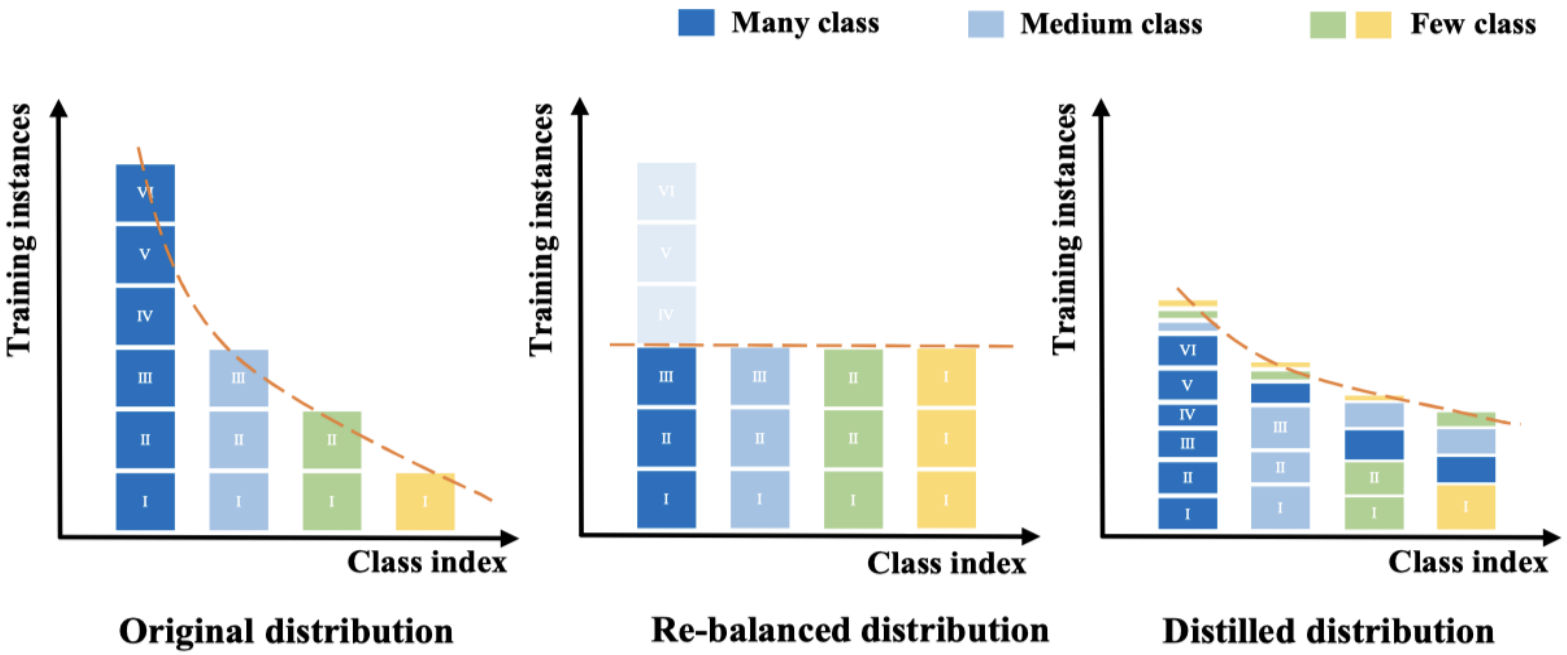


Figure 2: Illustration of the self-supervised task that we propose for semantic feature learning. Given four possible geometric transformations, the 0, 90, 180, and 270 degrees rotations, we train a ConvNet model $F(.)$ to recognize the rotation that is applied to the image that it gets as input. $F^y(X^{y^*})$ is the probability of rotation transformation y predicted by model $F(.)$ when it gets as input an image that has been transformed by the rotation transformation y^*.

Knowledge distillation

(1, 0, 0, 0)

(0.7, 0.1, 0.1, 0.1)



Soft labels are able to capture the inherent relation between classes

Self Supervision to Distillation for Long-Tailed Visual Recognition

Tianhao Li Limin Wang✉ Gangshan Wu
State Key Laboratory for Novel Software Technology, Nanjing University, China

ICCV 2021

Overall Framework

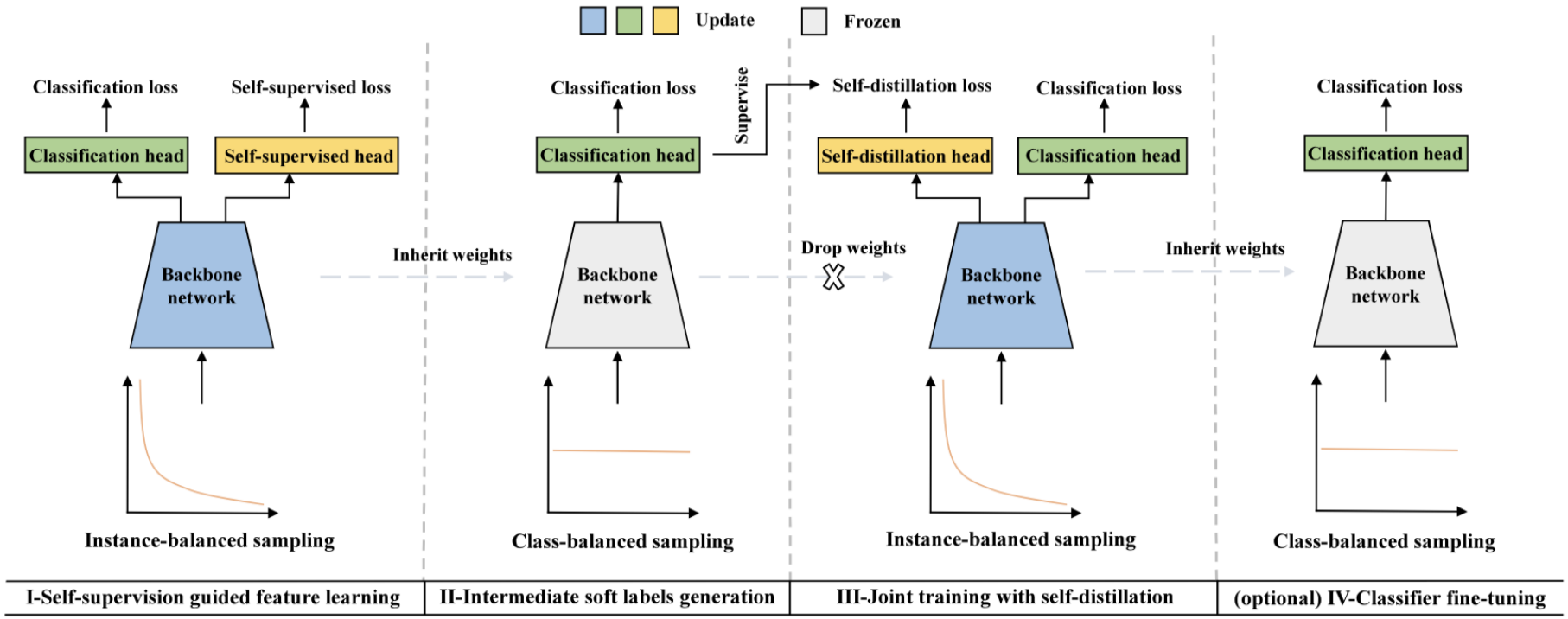


Figure 2. The pipeline of our Self Supervision to Distillation (SSD) framework. First, we train an initial feature network under label supervision and self-supervision jointly using instance-balanced sampling. Then, we refine the class decision boundaries with class-balanced sampling to generate soft labels by fixing the feature backbone. Finally, we train a self-distillation network with two classification heads under the supervision of both soft labels from previous stages and hard labels from the original training set.

Phase-I : Feature Learning

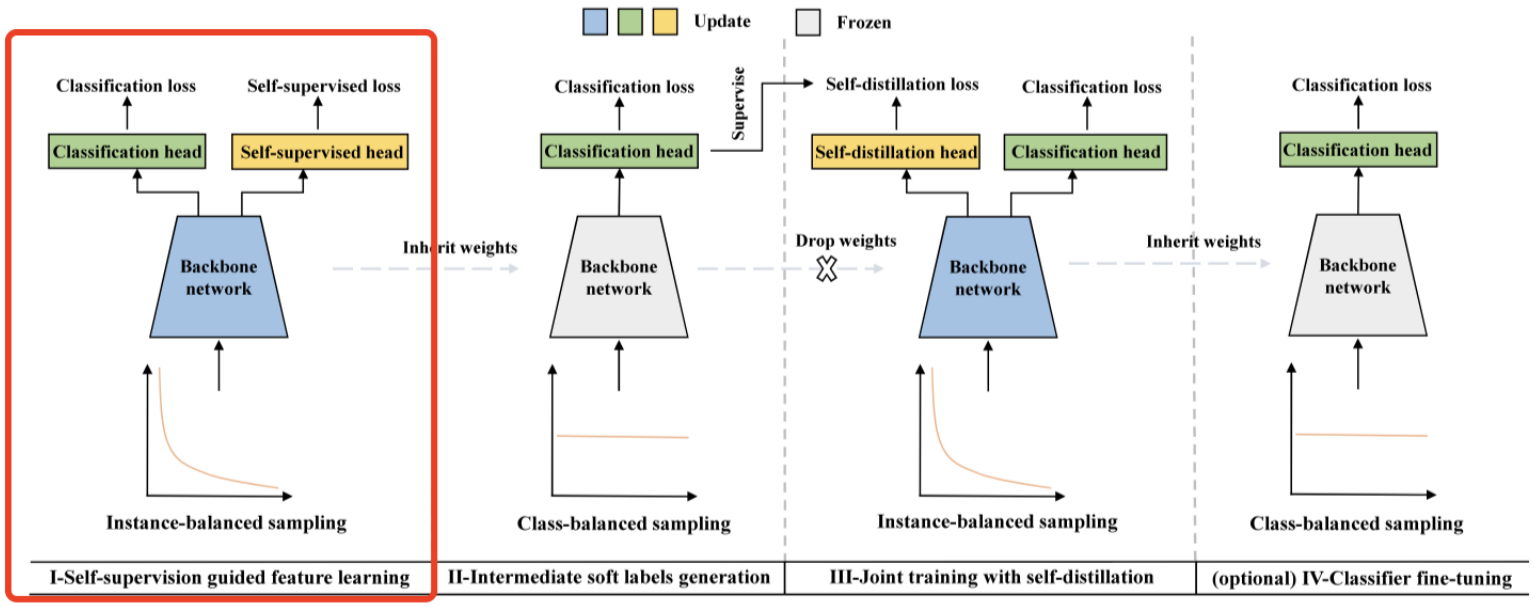


Figure 2. The pipeline of our Self Supervision to Distillation (SSD) framework. First, we train an initial feature network under label supervision and self-supervision jointly using instance-balanced sampling. Then, we refine the class decision boundaries with class-balanced sampling to generate soft labels by fixing the feature backbone. Finally, we train a self-distillation network with two classification heads under the supervision of both soft labels from previous stages and hard labels from the original training set.

The total loss of this stage is illustrated as:

$$\mathcal{L} = \alpha_1 \mathcal{L}_{sup}(\mathbf{x}; \theta, \omega_{sup}) + \alpha_2 \mathcal{L}_{self}(\mathbf{x}, \mathbf{y}; \theta, \omega_{self}),$$

Phase-{I, II} : Feature Learning

The total loss of this stage is illustrated as:

$$\mathcal{L} = \alpha_1 \mathcal{L}_{sup}(\mathbf{x}; \theta, \omega_{sup}) + \alpha_2 \mathcal{L}_{self}(\mathbf{x}, \mathbf{y}; \theta, \omega_{self}),$$

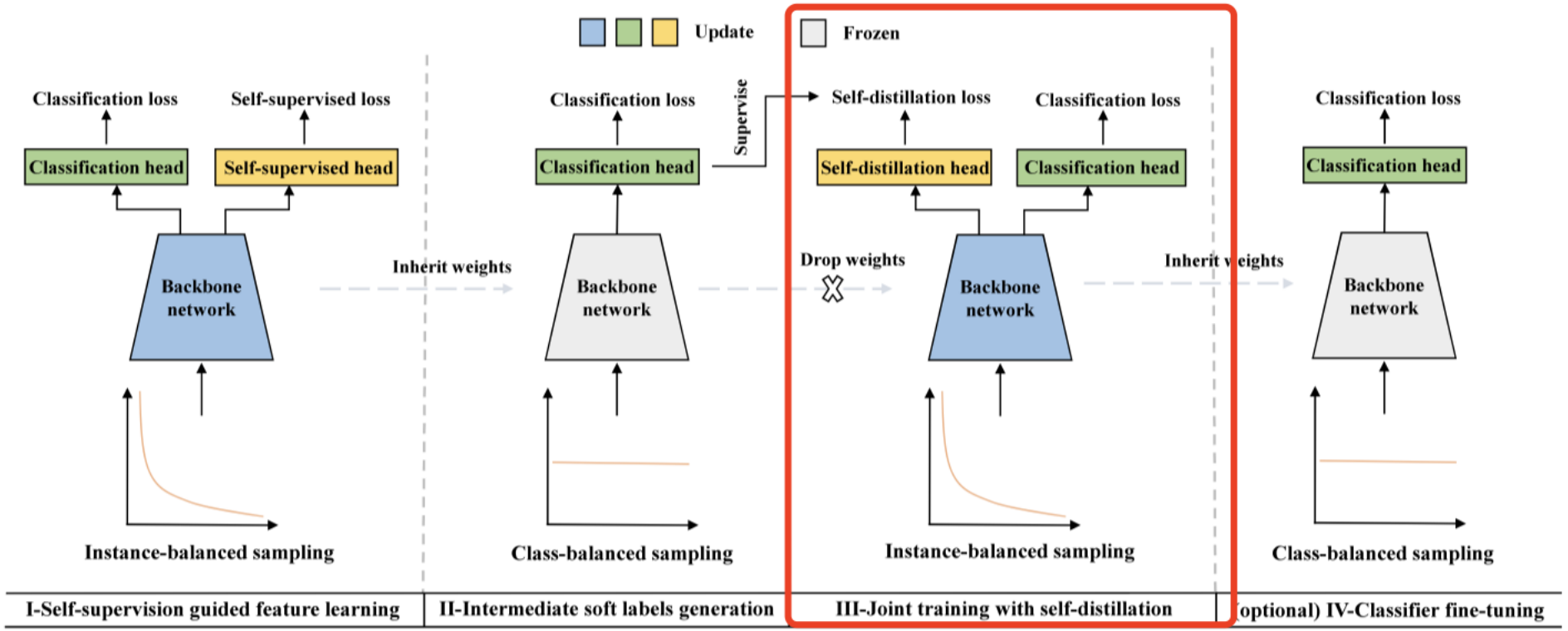
$\mathcal{L}_{sup}$ is Cross-entropy

moco v2(Rotation prediction)

$$\mathcal{L}_{self} = -\log\left(\frac{\exp(\mathbf{v}_i \mathbf{v'}_i/\tau)}{\exp(\mathbf{v}_i \mathbf{v'}_i/\tau) + \sum_K \exp(\mathbf{v}_i \mathbf{v'}_k/\tau)}\right),$$

α_1 and α_2 are hyper-parameters and equal to 1

Phase-III : Self-distillation



Joint training with self-distillation loss: $\mathcal{L} = \lambda_1 \mathcal{L}_{ce}(\mathbf{y}, \mathbf{z}^{hard}) + \lambda_2 \mathcal{L}_{kd}(\widetilde{\mathbf{y}}, \mathbf{z}^{soft})$

Phase-III : Self-distillation

Soft label:

$$\widetilde{y}_i = \frac{\exp(\widetilde{z}_i/T)}{\sum_{k=1}^{C}\exp(\widetilde{z}_k/T)}$$

$\tilde{z}$ denote the output logits of teacher model

The knowledge distillation loss:

$$\mathcal{L}_{kd}(\widetilde{\mathbf{y}}, \mathbf{z}^{soft}) = -T^2 \sum_{i=1}^{C} \widetilde{y}_i \log(\frac{\exp(z_i^{soft}/T)}{\sum_{k=1}^{C}\exp(z_k^{soft}/T)})$$

Joint training with self-distillation loss: $\mathcal{L} = \lambda_1 \mathcal{L}_{ce}(\mathbf{y}, \mathbf{z}^{hard}) + \lambda_2 \mathcal{L}_{kd}(\widetilde{\mathbf{y}}, \mathbf{z}^{soft})$

Experiment

Experiment

Methods	Imbalance factor		
	100	50	10
Cross Entropy (CE)*	39.1	44.0	55.8
Focal [27]	38.4	44.3	55.8
LDAM-DRW [2]	42.0	46.6	58.7
LWS* [20]	42.3	46.0	58.1
CE-DRW [48]	41.5	45.3	58.2
CE-DRS [48]	41.6	45.5	58.1
BBN [48]	42.6	47.0	59.1
M2m [23]	43.5	-	57.6
LFME [41]	43.8	-	-
Domain Adaption [19]	44.1	49.1	58.0
De-confound [35]	44.1	50.3	59.6
SSD (ours)	**46.0**	**50.5**	**62.3**

CIFAR100-LT

Methods	Top-1 Acc.	
	1×	2×
CB-Focal [2]	61.1	-
LDAM [2]	64.6	-
LDAM+DRW [2]	68.0	-
LDAM+DRW† [2]	64.6	66.1
τ-norm‡ [20]	65.6	69.3
cRT‡ [20]	65.2	68.5
LWS‡ [20]	65.9	69.5
CE-DRW [48]	63.7	-
CE-DRS [48]	63.6	-
BBN [48]	66.3	69.6
FSA [6]	65.9	-
LWS‡* [20]	66.6	69.5
SSD (ours)‡	**69.3**	**71.5**

iNaturalist 2018

Ablation studies

Ablation studies

Self-supervision guided feature learning:

Methods	1.5×	I	II	III-hard (test)	III-soft (test)	IV-LWS	Many	Medium	Few	Overall
CE							66.9	38.0	8.1	45.1
	✓						67.9	39.5	9.5	46.3
LWS							61.1	48.0	31.5	50.7
	✓						63.4	48.6	32.3	52.1
Our SSD	✓	✓					69.8	42.8	11.0	48.9
	✓	✓	✓				64.9	51.1	34.0	54.1
	✓		✓			✓	66.0	50.8	34.2	54.4
	✓	✓	✓	✓			**71.1**	46.1	15.6	51.6
	✓	✓	✓		✓		67.1	52.8	33.3	55.7
	✓	✓	✓			✓	66.8	**53.1**	**35.4**	**56.0**

Table 4. Ablation study on ImageNet-LT. We investigate the effectiveness of each stage of our proposed SSD method. Different stage are marked by Roman numerals I, II, III. The outputs of hard classifier and soft classifier are termed as III-hard and III-soft. IV-LWS means an extra classifier fine-tuning stage by LWS after self-distillation.

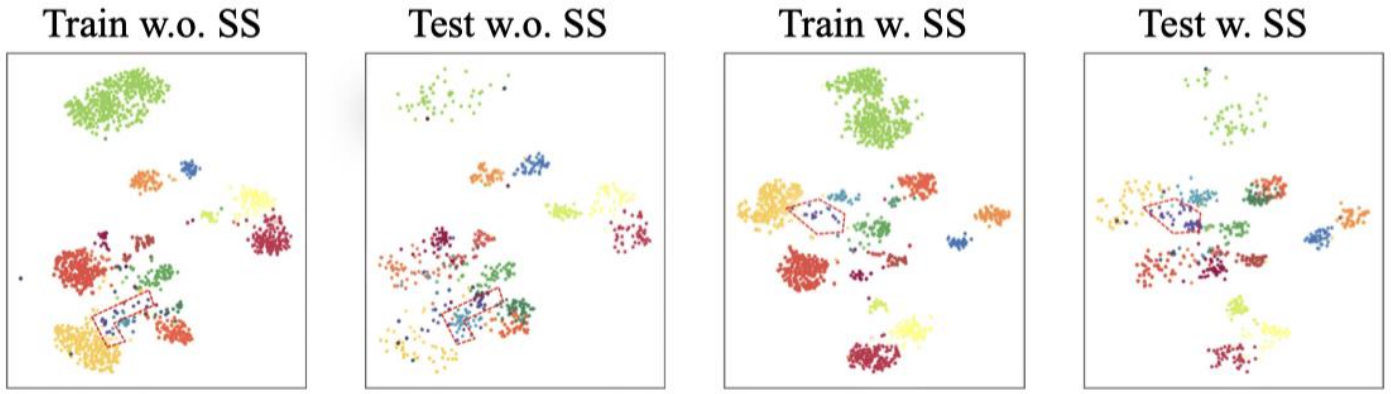


Figure 3. Visualization of self-supervision guided feature learning.

Ablation studies

Long-tailed recognition via self-distillation

Methods	1.5×	I	II	III-hard (test)	III-soft (test)	IV-LWS	Many	Medium	Few	Overall
CE							66.9	38.0	8.1	45.1
	✓						67.9	39.5	9.5	46.3
LWS							61.1	48.0	31.5	50.7
	✓						63.4	48.6	32.3	52.1
Our SSD	✓	✓					69.8	42.8	11.0	48.9
	✓	✓	✓				64.9	51.1	34.0	54.1
	✓		✓			✓	66.0	50.8	34.2	54.4
	✓	✓	✓	✓			**71.1**	46.1	15.6	51.6
	✓	✓	✓		✓		67.1	52.8	33.3	55.7
	✓	✓	✓			✓	66.8	**53.1**	**35.4**	**56.0**

Table 4. Ablation study on ImageNet-LT. We investigate the effectiveness of each stage of our proposed SSD method. Different stage are marked by Roman numerals I, II, III. The outputs of hard classifier and soft classifier are termed as III-hard and III-soft. IV-LWS means an extra classifier fine-tuning stage by LWS after self-distillation.

Ablation studies

Study on different self-distillation strategies:

Methods	Many	Medium	Few	Overall
Plain	67.9	39.5	9.5	46.3
Teacher model	64.9	51.1	**34.0**	54.1
Coupled	68.6	49.1	23.8	53.2
Single	67.4	52.0	31.3	55.1
Our III-hard	**71.1**	46.1	15.6	51.6
Our III-soft	67.1	**52.8**	33.3	**55.7**

Evaluation on self-supervised task:

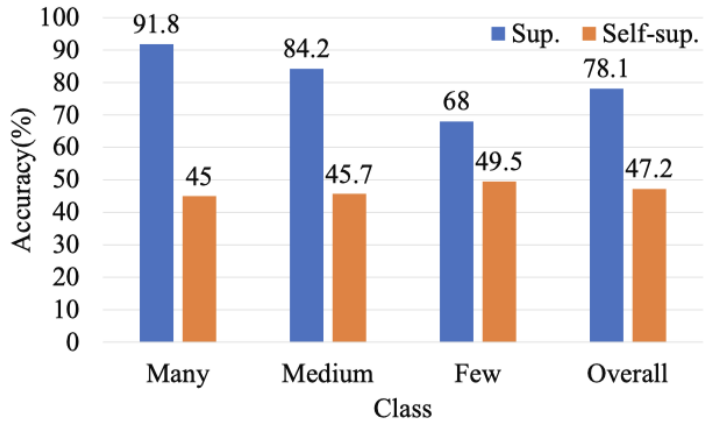


Figure 4. Training top-1 accuracy for supervised and self-supervised tasks for many-shot, medium-shot, few-shot and overall classes on the ImageNet-LT dataset.

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

ICCV 2021

Motivation

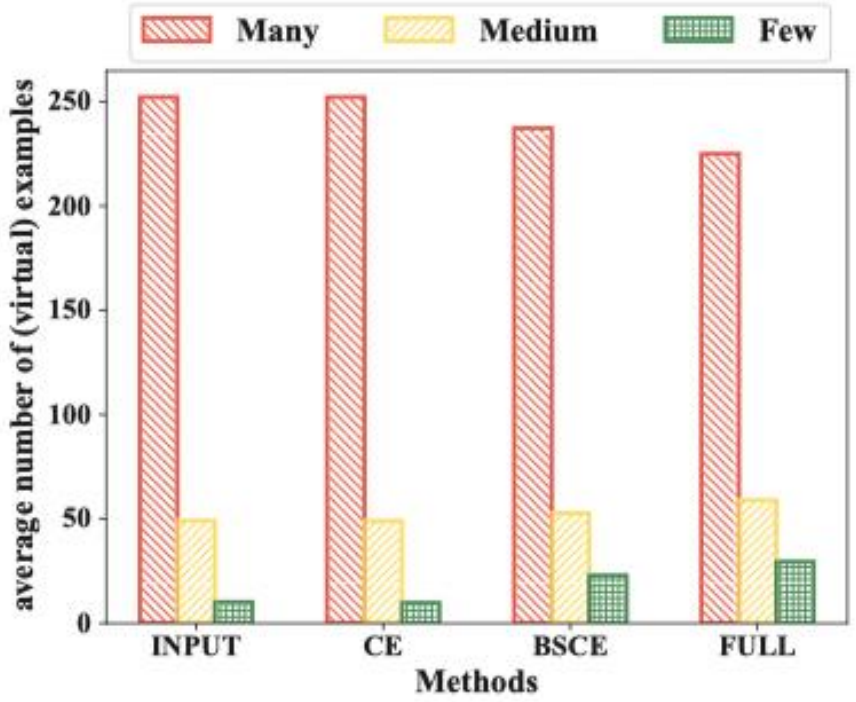


Figure 1. (Virtual) example distribution of different models.

Teacher: CIFAR-100-LT

- INPUT: Original input images
- CE: Cross entropy without specific tips
- BSCE: a long-tailed recognition method
- FULL: using CIFAR-100

Student: CIFAR-100-LT

- 39.2%
- 43.25%
- 53.71%

$KD \approx DLDL(\boldsymbol{\tau}=\mathbf{1})$

Suppose a teacher CNN model predicts t:

$$t_i = \frac{\exp(z_i)}{\sum_{k=1}^{C}\exp(z_k)}\,. \tag{1}$$

The student's loss function is:

$$L_{\mathrm{KD}} = (1-\alpha)L_{\mathrm{CE}}(\boldsymbol{y},\boldsymbol{s}) + \alpha L_{\mathrm{KL}}(\boldsymbol{t},\boldsymbol{s})\,. \tag{2}$$

First term:

$$L_{\mathrm{CE}}(\boldsymbol{y},\boldsymbol{s}) = -\sum_{k=1}^{C} y_k \log s_k\,. \tag{3}$$

Second term:

$$L_{\mathrm{KL}}(\boldsymbol{t},\boldsymbol{s}) = \sum_{k=1}^{C} t_k \log\frac{t_k}{s_k}\,. \tag{4}$$

$$t_i^{\tau} = \frac{\exp(z_i/\tau)}{\sum_{k=1}^{C}\exp(z_k/\tau)}\,, \tag{5}$$

$H(\cdot)$ to denote the entropy, define:

$$\tilde{\boldsymbol{t}} = (1-\alpha)\boldsymbol{y} + \alpha\boldsymbol{t}\,, \tag{6}$$

$$L_{CE}(x,y) = L_{KL}(x,y) + H(x)$$

$$H(\cdot) = -\sum_{i=1}^{N} P(x_i)\cdot logP(x_i)$$

$$\begin{aligned}
L_{\mathrm{KD}} &= (1-\alpha)L_{\mathrm{CE}}(\boldsymbol{y},\boldsymbol{s}) + \alpha L_{\mathrm{KL}}(\boldsymbol{t},\boldsymbol{s}) && (7)\\
&= (1-\alpha)L_{\mathrm{CE}}(\boldsymbol{y},\boldsymbol{s}) + \alpha L_{\mathrm{CE}}(\boldsymbol{t},\boldsymbol{s}) - \alpha H(\boldsymbol{t}) && (8)\\
&= L_{\mathrm{CE}}\big((1-\alpha)\boldsymbol{y} + \alpha\boldsymbol{t},\boldsymbol{s}\big) - \alpha H(\boldsymbol{t}) && (9)\\
&= L_{\mathrm{CE}}(\tilde{\boldsymbol{t}},\boldsymbol{s}) - \alpha H(\boldsymbol{t}) && (10)\\
&= L_{\mathrm{KL}}(\tilde{\boldsymbol{t}},\boldsymbol{s}) + H(\tilde{\boldsymbol{t}}) - \alpha H(\boldsymbol{t})\,. && (11)
\end{aligned}$$

The virtual example distribution must be flat

Binary classification example (airplane vs automobile):

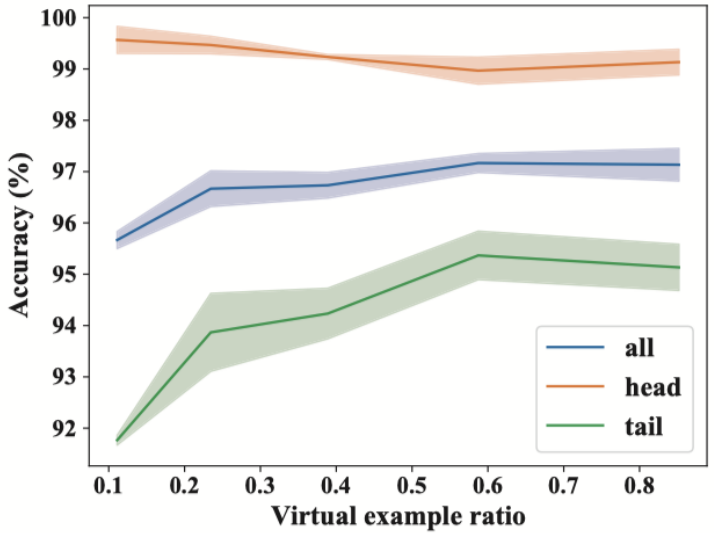
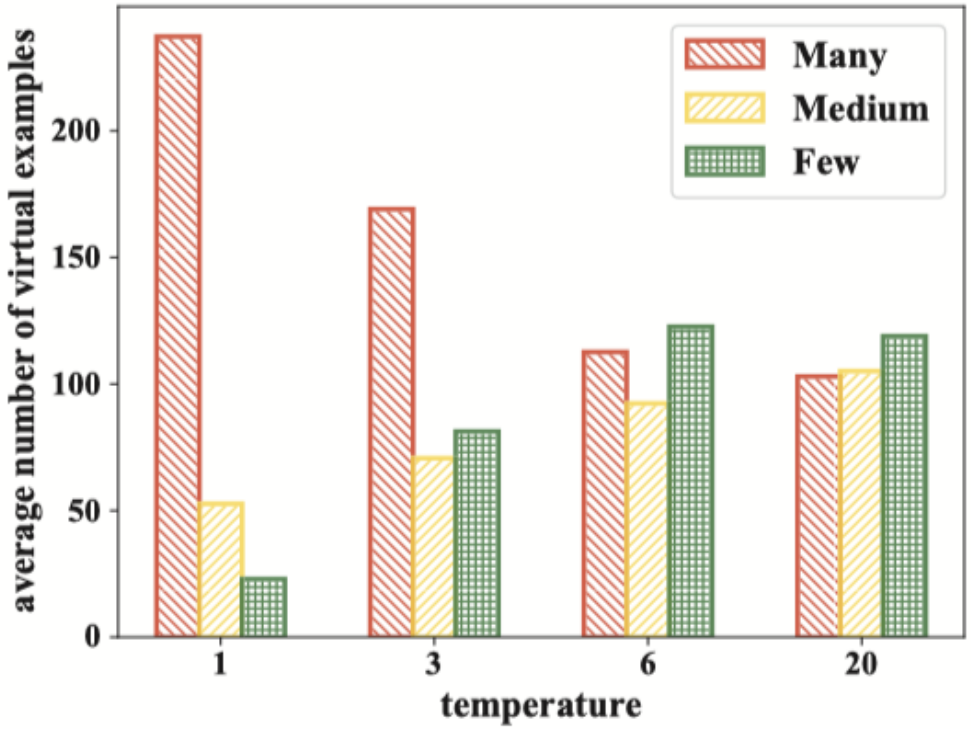


Figure 3. Accuracy (mean value and standard deviation) in a binary classification example. The accuracy becomes higher when the virtual example ratio between two classes grows (i.e., when the virtual example distribution becomes flatter). 'All' is the union of 'head' and 'tail'.



the head categories will help recognizing examples from the tail categories, even if these categories are not correlated

We prefer distributions that are flat, specifically, whose average number of examples per category in the tail part is slightly higher than that in the head part.

Level and distill virtual examples

Total loss function:

$$L_{\mathrm{DiVE}}(\boldsymbol{y}, \boldsymbol{s}^{\mathrm{BSCE}}) = (1-\alpha) L_{\mathrm{CE}}(\boldsymbol{y}, \boldsymbol{s}^{\mathrm{BSCE}}) + \alpha \tau^2 L_{\mathrm{KL}}(\boldsymbol{t}^{\tau}, \boldsymbol{s}^{\tau}) .$$

t^{τ} uses a temperature τ and possibly followed by a power normalization ($p = 0.5$)

$$t_k^{\tau} \longleftarrow \sqrt{t_k^{\tau}}, \qquad \forall\, 1 \le k \le C , \qquad (13)$$

$$t_i^{\tau} \longleftarrow \frac{t_i^{\tau}}{\sum_k t_k^{\tau}} \qquad \forall\, 1 \le i \le C . \qquad (14)$$

s^{τ} only uses the temperature τ and does not apply the power normalization.

Experiment

Experimental Results

Table 1. Top-1 accuracy (%) on CIFAR-100-LT. The "†" symbol denotes results copied directly from [36].

Methods	Imbalance factor		
	100	50	10
CE	38.35	42.41	56.51
Focal† [18]	38.41	44.32	55.78
BSCE	42.39	47.60	58.38
LFME [31]	43.80	-	-
LDAM-DRW [1]	42.04	46.62	58.71
BBN [36]	42.56	47.02	59.12
Meta-learning [16]	44.70	50.08	59.59
LDAM-DRW+SSP [33]	43.43	47.11	58.91
TDE [28]	44.10	50.30	59.60
DiVE	**45.35**	**51.13**	**62.00**

CIFAR-100-LT

Experimental Results

Methods	Many	Medium	Few	All
CE	**65.02**	37.07	8.07	43.89
BSCE	60.92	47.97	29.79	50.48
OLTR† [19]	-	-	-	46.30
τ-norm [17]	59.10	46.90	30.70	49.40
LWS [17]	60.20	47.20	30.30	49.90
TDE [28]	62.70	48.80	**31.60**	51.80
TDE*	62.56	47.83	29.91	51.06
DiVE	64.06	**50.41**	31.46	**53.10**

ImageNet-LT

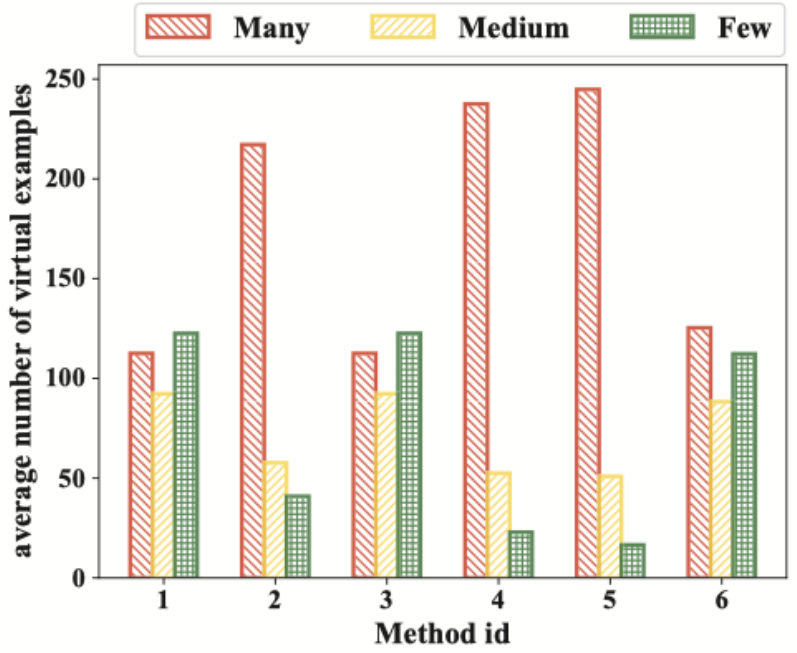
Methods	90 epochs		200 epochs	
	top-1	top-5	top-1	top-5
CE	62.60	83.44	-	-
CB-Focal† [3]	61.12	81.03	-	-
BSCE	65.35	83.36	67.84	85.45
LDAM-DRW† [1]	68.00	85.18	-	-
BBN [36]	66.29	85.57	69.65	87.64
Meta-learning [16]	67.55	86.17	-	-
LWS [17]	65.90	-	69.50	-
cRT+SSP [33]	68.10	-	-	-
DiVE	**69.13**	**86.85**	**71.71**	**88.39**

iNaturalist2018

Further Analyses

Table 6. Effects of balancing the virtual example distribution.

	BSCE	$\tilde{t}/t^{\tau}$	τ	power	100	50	10
CE	-	-	-	-	38.35	42.41	56.51
# 1	✓	t^{τ}	3	✓	**45.35**	**51.13**	**62.00**
# 2	✗	$\tilde{t}$	1	✓	44.55	49.69	61.62
# 3	✗	t^{τ}	3	✓	44.50	50.20	61.28
# 4	✗	t^{τ}	1	✗	43.25	47.64	60.07
# 5	✗	$\tilde{t}$	1	✗	41.59	47.10	59.10
# 6	✗	$\tilde{t}$	3	✓	43.22	48.51	60.59



THANKS