Distribution Alignment: A Unified Framework for Long-tail Visual Recognition (Supplementary Material)

Songyang Zhang[1,3,5,*] Zeming Li[2] Shipeng Yan[1] Xuming He[1,4] Jian Sun[2]
[1]ShanghaiTech University [2]Megvii Technology [3] University of Chinese Academy of Sciences
[4]Shanghai Engineering Research Center of Intelligent Vision and Imaging
[5]Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences
{zhangsy1, yanshp, hexm}@shanghaitech.edu.cn, {lizeming,sunjian}@megvii.com

1. Experiments of Image Classification

In this section, we first introduce the dataset and evaluation metrics for image classification task in Sec.1.1. Then the training configuration will be detailed in Sec.1.2, followed by results on three benchmarks in Sec.1.3.

1.1. Dataset and Evaluation Metrics

Datasets To demonstrate our methods, we conduct experiments on three large-scale long-tailed datasets, including Places-LT[7], ImageNet-LT[7], and iNaturalist 2018[11]. Places-LT and ImageNet-LT are artificially generated by sampling a subset from their balanced versions (Places-365[7] and ImageNet-2012[2]) following the *Parento distribution*. iNaturalist 2018 is a real-world, naturally long-tailed dataset, consisting of samples from 8,142 species.

Evaluation Metrics We report the class-balanced average *Top-1* accuracy on the corresponding validation/test set, and also calculate the accuracy of three disjoint subsets, 'Many', 'Medium' and 'Few', which are defined according to the amount of training data per class [4].

1.2. Training Configuration

Configuration Detail Following [4], we use PyTorch[9] framework for all experiments. For *ImageNet-LT*, we report performance with ResNet-{50,101,152} and ResNeXt-{50,101,152} and mainly use ResNet-50 for ablation study. For *iNaturalist 2018*, performance is reported with ResNet-{50,101,152}. For *Places-LT*, ResNet-152 is used as backbone and we pre-train it on the full ImageNet-2012 dataset.

We use the SGD optimizer with momentum 0.9, batch size 256, cosine learning rate schedule gradually decaying from 0.1 to 0, and image resolution 224×224. For the joint learning stage, the backbone network and original classifier head are jointly trained with 90 epochs for ImageNet-LT, and 90/200 epochs for iNaturalist-2018. For the Places-LT dataset, the models are trained with 30 epochs with the all layers frozen expect the last ResNet block in the first stage.

Implementation of Our Method In the second distribution alignment stage, we restart the learning rate and train it for 10/30 epochs as [4] while keeping the backbone network and original classifier head fixed(10 epochs for ImageNet-LT and Places-LT, 30 epochs for iNaturalist-2018). For all three datasets, we set the generalized re-weight scale $\rho = 1.2$ for dot-product classifier head, $\rho = 1.5$ for cosine normalized classifier head. The α and β are initialized with 1.0 and 0.0, respectively.

1.3. Detailed Experimental Results

ImageNet-LT. We present the detailed quantitative results for ImageNet-LT in Table 1.

iNaturalist and Places-LT. To further demonstrate our method, we conduct experiments on two extra large-scale long-tail benchmarks and report the performance in Table 4 and Table 5.

1.4. Ablation Study

Influence of Model Components We report an ablation study of the two main components of our method with ResNeXt-50 in Tab. 2, which shows that both adaptive calibration and generalized re-weighting(G-RW) contribute to the performance improvement of our approach.

Analysis of the Calibration We plot the learned magnitude and margin according to the class sizes below. They share a similar trend, in which the tail/body classes have

[*]This work was done when Songyang Zhang was a research intern at Megvii Technology. This work was supported by Shanghai NSF Grant (No. 18ZR1425100), National Key R&D Program of China (No. 2017YFA0700800), and Beijing Academy of Artificial Intelligence (BAAI). Code is available: https://github.com/Megvii-BaseDetection/DisAlign

Backbone	Method	ResNet				ResNeXt			
		Average	Many	Medium	Few	Average	Many	Medium	Few
*-50	Baseline	41.6	64.0	33.8	5.8	44.4	65.9	37.5	7.7
	Baseline*	48.4	68.4	41.7	15.2	49.2	68.9	42.8	15.6
	DisAlign	51.3	59.9	49.9	31.8	52.6	61.5	50.7	33.1
	DisAlign*	52.9	61.3	52.2	31.4	53.4	62.7	52.1	31.4
*-101	Baseline	44.2	66.6	36.8	7.1	44.8	66.2	37.8	8.6
	Baseline*	49.5	69.3	43.1	15.9	50.0	69.9	43.7	15.9
	DisAlign	52.7	61.7	51.1	32.4	53.6	63.3	51.2	34.6
	DisAlign*	54.1	63.2	53.1	31.9	54.6	64.7	53.0	31.7
*-152	Baseline	44.9	66.9	37.7	7.7	47.8	69.1	41.4	10.4
	Baseline*	50.2	70.1	43.9	16.1	50.5	70.0	44.4	16.5
	DisAlign	53.7	62.8	51.9	34.2	54.5	64.5	52.0	34.7
	DisAlign*	54.8	63.9	53.9	32.5	55.0	65.1	53.3	32.2

Table 1: **Top-1 Accuracy on ImageNet-LT test set.** All models use the feature extractor and original classifier head trained with 90 epoch in joint learning stage, $*$ denotes the model uses cosine classifier head.

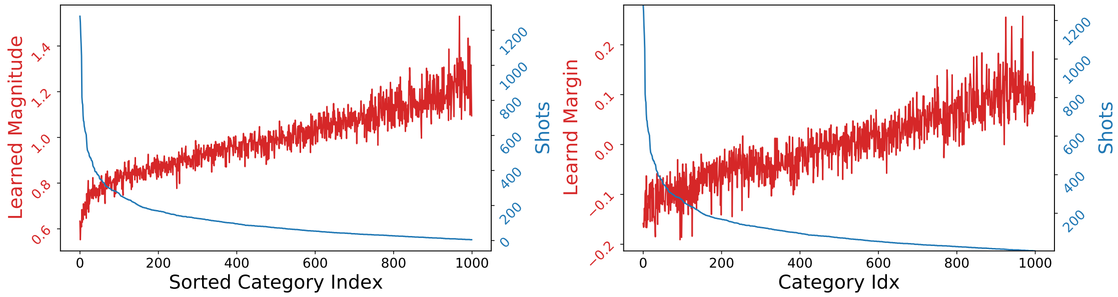

Figure 1: **Analysis of the Calibration.** We use model trained on ImageNet-LT with ResNeXt-50 for analysis.

Method	Calibration	G-RW	Top-1 Acc
Baseline*	-	-	49.2
cRT*	✗	✗	49.7
-	✗	✓	51.9
DisAlign*	✓	✓	53.4

Table 2: **Influence of Model Components.** Backbone is ResNeXt-50, $*$ means cosine classifier.

larger value than head. Thus our calibration alleviates the bias in the original prediction by boosting the tail scores.

Confidence Score We study confidence-based calibration in the table below, which shows that the input-aware calibration outperforms the input-agnostic counterpart and the baselines using only magnitude or margin. We also observe that the example whose biased prediction probability is low on its ground-truth class tends to be improved with higher confidence.

Generalized Re-weighting		✓	✓	✓	✓	✓	✓
Magnitude(w/o Confidence)		✓				✓	
Magnitude			✓				✓
Margin(w/o Confidence)				✓		✓	
Margin					✓		✓
Average Accuracy	41.6	49.9	50.1	49.6	49.9	51.0	51.3

Table 3: **Ablation of the Confidence Score.** We extend the Tab.5(main paper) to analyze the influence of confidence score.

2. Experiments of Semantic Segmentation

Similar to image classification, the large-scale semantic segmentation task still suffers from the long-tail data distribution. To further validate the effectiveness of our method, we also apply DisAlign on large-scale semantic segmentation benchmark: ADE-20k.

Backbone	Method	90 Epoch				200 Epoch			
		Average	Many	Medium	Few	Average	Many	Medium	Few
ResNet-50	Baseline	61.7	72.2	63.0	57.2	65.8	75.7	66.9	61.7
	Baseline*	64.8	75.8	66.6	59.7	66.2	77.3	68.3	60.7
	DisAlign	67.8	64.1	68.5	67.9	70.6	69.0	71.1	70.2
	DisAlign*	69.5	61.6	70.8	69.9	70.2	68.0	71.3	69.4
ResNet-101	Baseline	64.6	75.9	66.0	59.9	67.3	75.5	68.9	63.2
	Baseline*	66.4	76.8	68.5	61.1	68.0	78.9	69.7	63.0
	DisAlign	70.0	68.3	70.4	69.9	72.9	73.0	73.5	72.1
	DisAlign*	70.8	65.4	72.2	70.4	71.9	69.3	72.6	71.8
ResNet-152	Baseline	65.0	75.2	66.3	60.7	69.0	78.2	70.6	64.7
	Baseline*	67.3	77.8	69.4	61.8	69.0	78.5	71.0	64.0
	DisAlign	71.3	70.7	71.8	70.8	74.1	74.9	74.4	73.5
	DisAlign*	71.7	67.1	73.0	71.3	72.8	70.6	73.6	72.3

Table 4: Top-1 Accuracy on iNaturalist 2018 with different backbones(ResNet-{50,101,152}) and different training epochs(90 & 200), $*$ denotes the model uses cosine classifier head.

Backbone	Method	Top-1 Accuracy			
		Average	Many	Medium	Few
R-50	Baseline	29.2	45.3	25.5	8.0
	DisAlign	37.8	39.3	40.7	28.5
R-101	Baseline	30.2	46.1	26.9	8.4
	DisAlign	38.5	39.1	42.0	29.1
R-152	Baseline	30.2	45.7	27.3	8.2
	DisAlign	39.3	40.4	42.4	30.1

Table 5: Top-1 Accuracy on Places-LT with different backbones(ResNet-{50,101,152}).

2.1. Dataset and Evaluation

Dataset. ADE20K dataset is a scene parsing benchmark, which contains 150 stuff/object categories. The dataset includes 20K/2K/3K images for training, validation, and testing. Compared with the image classification[7], the imbalance of ADE20K is more serve than the image classification, which has an imbalance ratio of **788**(Max/Min). Follow the similar protocol in image classification, we divide the 150 categories into 3 groups according to the ratio of pixel number over the whole dataset. Specifically, three disjoint subsets are: *head classes*(classes each with a ratio over 1.0%), *body classes*(classes each with a ratio ranging from 0.1% to 1%) and *tail classes*(classes under a ratio of 0.1%), the complete list of the split is reported in Tab.7.

Evaluation. For the evaluation metric, we use the mean intersection of union(mIoU) and mean pixel accuracy(mAcc). We also report the mIoU and mAcc of each group(head, body and tail) for clarity.

2.2. Training Configuration

We implement our method based on MMSegmentation toolkit[8]. In the joint learning training phase, we set the learning rate to 0.01 initially, which gradually decreases to 0 by following the 'poly' strategy as [13]. The images are cropped to 512 × 512 and augmented with randomly scaling(from 0.5 to 2.0) and flipping. ResNet-50, ResNet-101 and ResNeSt-101[14] are used as the backbone. For the evaluation metric, we use the mean intersection of union(mIoU) and mean pixel accuracy(mAcc). All models are trained with 160k iterations with a batch size of 32 based on 8 V100 GPUs. In the DisAlign stage, we follow a similar protocol as stage-1 and only training the model with 8k iterations. We set $\rho = 0.3$ for all experiments.

2.3. Quantitative Results

We evaluate our method with two state-of-the-art segmentation models(FCN[10] and DeepLabV3+[1])based on different backbone networks, ranging from ResNet-50, ResNet-101 to the latest ResNeSt-101, and report the performance in Tab.6.

3. Experiments on LVIS Dataset

3.1. Dataset and Evaluation Protocol

Dataset. LVIS v0.5[3] dataset is a benchmark dataset for research on large vocabulary object detection and instance segmentation, which contains 56K images over 1230 categories for training, 5K images for validation. This challenging dataset is an appropriate benchmark to study the large-scale long-tail problem, where the categories can be binned into three types similar with ImageNet-LT: *rare*(1-10 training images), *common*(11-100 training images), and

Framework	B	Method	Aug	Mean IoU				Mean Accuracy			
				Average	Head	Body	Tail	Average	Head	Body	Tail
FCN[10]	R-50	Baseline	✗	36.1	62.5	38.1	27.6	45.4	76.9	48.8	34.5
		DisAlign	✗	37.5(+1.4)	62.6(+0.1)	40.2(+2.1)	28.8(+1.2)	49.9(+4.5)	76.7(-0.2)	54.9(+6.1)	39.0(+4.5)
		Baseline	✓	38.1	64.6	40.0	29.6	46.3	78.6	49.3	35.4
		DisAlign	✓	40.1(+2.0)	65.0(+0.4)	42.8(+2.8)	31.3(+1.7)	51.4(+5.1)	78.6(+0.0)	56.1(+6.8)	40.6(+5.2)
	R-101	Baseline	✗	39.9	65.3	42.0	31.7	49.6	79.1	52.6	39.6
		DisAlign	✗	41.8(+1.9)	65.5(+0.2)	44.1(+2.1)	33.7(+2.0)	54.7(+5.1)	79.0(-0.1)	58.6(+6.0)	45.2(+5.6)
		Baseline	✓	41.4	67.0	43.3	33.2	50.2	80.6	52.9	40.1
		DisAlign	✓	43.7(+2.3)	67.4(+0.4)	46.1(+2.8)	35.7(+2.5)	55.9(+5.7)	80.6(+0.0)	59.7(+6.8)	46.4(+6.3)
	S-101	Baseline	✗	45.6	66.6	47.5	38.6	57.8	78.8	62.1	48.9
		DisAlign	✗	46.2(+0.6)	66.6(+0.0)	48.0(+0.4)	39.4(+0.8)	60.3(+2.5)	79.1(+0.3)	64.9(+2.8)	51.7(+2.8)
		Baseline	✓	46.2	67.6	48.0	39.1	57.3	79.4	61.7	48.2
		DisAlign	✓	46.9(+0.7)	67.7(+0.1)	48.2(+0.2)	40.3(+1.2)	60.1(+2.8)	79.7(+0.3)	64.2(+2.5)	51.9(+3.7)
DeepLabV3^{+}[1]	R-50	Baseline	✗	43.9	66.6	47.1	35.6	54.9	79.4	60.3	44.5
		DisAlign	✗	44.4(+0.5)	66.6(+0.0)	47.2(+0.1)	36.5(+0.9)	57.2(+2.3)	79.8(+0.4)	62.3(+2.0)	47.5(+3.0)
		Baseline	✓	44.9	67.7	48.3	36.4	55.0	80.1	60.8	44.1
		DisAlign	✓	45.7(+0.8)	67.7(+0.0)	48.6(+0.3)	37.8(+1.4)	57.3(+2.3)	80.8(+0.7)	63.0(+2.2)	46.9(+2.8)
	R-101	Baseline	✗	45.5	67.6	48.2	37.6	56.4	80.1	61.2	46.6
		DisAlign	✗	46.0(+0.5)	67.6(+0.0)	48.4(+0.2)	38.5(+0.9)	59.1(+2.7)	80.5(+0.4)	63.8(+2.6)	49.9(+3.3)
		Baseline	✓	46.4	68.7	49.0	38.4	56.7	80.9	61.5	46.7
		DisAlign	✓	47.1(+0.7)	68.7(+0.0)	49.4(+0.4)	39.6(+1.2)	59.5(+2.8)	81.4(+0.5)	64.2(+2.7)	50.3(+3.6)
	S-101	Baseline	✗	46.5	68.0	49.1	38.8	58.1	80.1	63.4	48.5
		DisAlign	✗	46.9(+0.4)	67.8(-0.2)	49.2(+0.1)	39.6(+0.8)	60.7(+2.6)	80.5(+0.4)	65.5(+2.1)	51.9(+3.4)
		Baseline	✓	47.3	69.0	49.7	39.7	58.1	80.8	63.4	48.2
		DisAlign	✓	47.8(+0.5)	68.9(-0.1)	49.8(+0.1)	40.7(+1.0)	60.1(+2.0)	81.0(+0.2)	65.5(+2.1)	52.0(+3.8)

Table 6: **Results on ADE-20K:** All baseline models are trained with a image size of 512x512 and 160K iteration in total. **Aug** denotes multi-scale is used for inference.

Category	Ratio	Group	Category	Ratio	Group	Category	Ratio	Group	Category	Ratio	Group	Category	Ratio	Group
'wall'	0.1576,	Head	'armchair'	0.0044,	Body	'river'	0.0015,	Body	'airplane'	0.0007,	Tail	'food'	0.0005,	Tail
'building'	0.1072,	Head	'seat'	0.0044,	Body	'bridge'	0.0015,	Body	'dirt track'	0.0007,	Tail	'step'	0.0004,	Tail
'sky'	0.0878,	Head	'fence'	0.0033,	Body	'bookcase'	0.0014,	Body	'apparel'	0.0007,	Tail	'tank'	0.0004,	Tail
'floor'	0.0621,	Head	'desk'	0.0031,	Body	'blind'	0.0014,	Body	'pole'	0.0006,	Tail	'trade name'	0.0004,	Tail
'tree'	0.048,	Head	'rock'	0.003,	Body	'coffee table'	0.0014,	Body	'land'	0.0006,	Tail	'microwave'	0.0004,	Tail
'ceiling'	0.045,	Head	'wardrobe'	0.0027,	Body	'toilet'	0.0014,	Body	'bannister'	0.0006,	Tail	'pot'	0.0004,	Tail
'road'	0.0398,	Head	'lamp'	0.0026,	Body	'flower'	0.0014,	Body	'escalator'	0.0006,	Tail	'animal'	0.0004,	Tail
'bed'	0.0231,	Head	'bathtub'	0.0024,	Body	'book'	0.0013,	Body	'ottoman'	0.0006,	Tail	'bicycle'	0.0004,	Tail
'windowpane'	0.0198,	Head	'railing'	0.0024,	Body	'hill'	0.0013,	Body	'bottle'	0.0006,	Tail	'lake'	0.0004,	Tail
'grass'	0.0183,	Head	'cushion'	0.0023,	Body	'bench'	0.0013,	Body	'buffet'	0.0006,	Tail	'dishwasher'	0.0004,	Tail
'cabinet'	0.0181,	Head	'base'	0.0023,	Body	'countertop'	0.0012,	Body	'poster'	0.0006,	Tail	'screen'	0.0004,	Tail
'sidewalk'	0.0166,	Head	'box'	0.0022,	Body	'stove'	0.0012,	Body	'stage'	0.0006,	Tail	'blanket'	0.0004,	Tail
'person'	0.016,	Head	'column'	0.0022,	Body	'palm'	0.0012,	Body	'van'	0.0006,	Tail	'sculpture'	0.0004,	Tail
'earth'	0.0151,	Head	'signboard'	0.002,	Body	'kitchen island'	0.0012,	Body	'ship'	0.0006,	Tail	'hood'	0.0004,	Tail
'door'	0.0118,	Head	'chest of drawers'	0.0019,	Body	'computer'	0.0011,	Body	'fountain'	0.0005,	Tail	'sconce'	0.0003,	Tail
'table'	0.011,	Head	'counter'	0.0019,	Body	'swivel chair'	0.001,	Tail	'conveyer belt'	0.0005,	Tail	'vase'	0.0003,	Tail
'mountain'	0.0109,	Head	'sand'	0.0018,	Body	'boat'	0.0009,	Tail	'canopy'	0.0005,	Tail	'traffic light'	0.0003,	Tail
'plant'	0.0104,	Head	'sink'	0.0018,	Body	'bar'	0.0009,	Tail	'washer'	0.0005,	Tail	'tray'	0.0003,	Tail
'curtain'	0.0104,	Head	'skyscraper'	0.0018,	Body	'arcade machine'	0.0009,	Tail	'plaything'	0.0005,	Tail	'ashcan'	0.0003,	Tail
'chair'	0.0103,	Head	'fireplace'	0.0018,	Body	'hovel'	0.0009,	Tail	'swimming pool'	0.0005,	Tail	'fan'	0.0003,	Tail
'car'	0.0098,	Body	'refrigerator'	0.0018,	Body	'bus'	0.0009,	Tail	'stool'	0.0005,	Tail	'pier'	0.0003,	Tail
'water'	0.0074,	Body	'grandstand'	0.0018,	Body	'towel'	0.0008,	Tail	'barrel'	0.0005,	Tail	'crt screen'	0.0003,	Tail
'painting'	0.0067,	Body	'path'	0.0018,	Body	'light'	0.0008,	Tail	'basket'	0.0005,	Tail	'plate'	0.0003,	Tail
'sofa'	0.0065,	Body	'stairs'	0.0017,	Body	'truck'	0.0008,	Tail	'waterfall'	0.0005,	Tail	'monitor'	0.0003,	Tail
'shelf'	0.0061,	Body	'runway'	0.0017,	Body	'tower'	0.0008,	Tail	'tent'	0.0005,	Tail	'bulletin board'	0.0003,	Tail
'house'	0.006,	Body	'case'	0.0017,	Body	'chandelier'	0.0008,	Tail	'bag'	0.0005,	Tail	'shower'	0.0003,	Tail
'sea'	0.0053,	Body	'pool table'	0.0017,	Body	'awning'	0.0007,	Tail	'minibike'	0.0005,	Tail	'radiator'	0.0003,	Tail
'mirror'	0.0052,	Body	'pillow'	0.0017,	Body	'streetlight'	0.0007,	Tail	'cradle'	0.0005,	Tail	'glass'	0.0002,	Tail
'rug'	0.0046,	Body	'screen door'	0.0015,	Body	'booth'	0.0007,	Tail	'oven'	0.0005,	Tail	'clock'	0.0002,	Tail
'field'	0.004	Body	'stairway'	0.0015	Body	'television receiver'	0.0007	Tail	'ball'	0.0005	Tail	'flag'	0.0002	Tail

Table 7: **Splits of ADE-20K:** The ratio of each category is reported according to [15].

Backbone	**Method**	**BBox AP**				**Mask AP**			
		$\mathbf{AP}_{bbox}$	$\mathbf{AP}^{r}_{bbox}$	$\mathbf{AP}^{c}_{bbox}$	$\mathbf{AP}^{f}_{bbox}$	$\mathbf{AP}_{mask}$	$\mathbf{AP}^{r}_{mask}$	$\mathbf{AP}^{c}_{mask}$	$\mathbf{AP}^{f}_{mask}$
ResNet-50	Baseline	20.8	3.3	19.5	29.4	21.2	3.7	21.6	28.4
	DisAlgin	23.9	7.5	25.0	29.1	24.2	8.5	26.2	28.0
	Baseline*	22.8	10.3	21.1	30.1	23.8	11.5	23.7	28.9
	DisAlgin*	25.6	13.7	25.6	30.5	26.3	14.9	27.6	29.2
ResNet-101	Baseline	22.2	2.6	21.1	31.6	22.6	2.7	22.8	30.2
	DisAlgin	25.6	9.0	26.5	30.9	25.8	10.3	27.6	29.6
	Baseline*	24.5	10.1	23.2	31.8	25.1	11.2	25.2	30.4
	DisAlgin*	27.5	15.9	27.6	32.0	28.2	17.8	29.7	30.5
ResNeXt-101	Baseline	24.5	3.9	24.1	33.1	25.0	4.2	26.3	31.8
	DisAlgin	26.8	8.8	27.6	33.0	27.4	11.0	29.3	31.6
	Baseline*	26.9	12.1	26.1	33.8	27.7	15.2	28.2	32.2
	DisAlgin*	29.5	17.7	29.5	33.8	30.0	19.6	31.5	32.3

Table 8: **Results on LVIS v0.5 dataset with Mask R-CNN.** * denotes the model use cosine classifier head.

Backbone	**Method**	**BBox AP**				**Mask AP**			
		$\mathbf{AP}_{bbox}$	$\mathbf{AP}^{r}_{bbox}$	$\mathbf{AP}^{c}_{bbox}$	$\mathbf{AP}^{f}_{bbox}$	$\mathbf{AP}_{mask}$	$\mathbf{AP}^{r}_{mask}$	$\mathbf{AP}^{c}_{mask}$	$\mathbf{AP}^{f}_{mask}$
ResNet-50	Baseline	25.2	3.7	24.3	34.8	23.0	3.5	23.0	30.8
	DisAlgin	28.7	9.0	30.2	34.6	26.1	8.4	28.1	30.7
	Baseline*	28.8	15.4	28.2	34.9	26.2	13.6	26.3	31.1
	DisAlgin*	32.2	21.6	33.3	35.2	29.4	19.4	30.9	31.4
ResNet-101	Baseline	26.1	3.4	25.4	35.9	24.0	3.3	24.2	32.0
	DisAlgin	29.7	8.1	31.7	35.8	27.3	7.8	29.7	32.0
	Baseline*	30.4	15.5	30.3	36.5	28.1	13.9	29.2	32.4
	DisAlgin*	33.7	22.1	34.9	36.9	30.9	19.0	33.2	32.8
ResNeXt-101	Baseline	28.4	4.6	28.6	37.5	26.1	4.6	27.2	33.4
	DisAlgin	31.3	9.5	33.2	37.7	28.7	9.0	31.1	33.6
	Baseline*	32.6	18.5	32.8	37.9	29.8	16.9	30.9	33.7
	DisAlgin*	34.7	24.6	35.3	38.1	31.8	22.0	33.2	33.9

Table 9: **Results on LVIS v0.5 dataset with Cascade R-CNN.** * denotes the model use cosine classifier head.

frequent($>$ 100 training images).

Evaluation Protocol. We evaluate our method on LVIS for object detection and instance segmentation. For evaluation, we use a COCO-style average precision(AP) metric that averages over categories and different box/mask intersection over union(IoU) threshold[6]. All standard LVIS evaluation metrics including AP, AP^r, AP^c, AP^f for box bounding boxes and segmentation masks. Subscripts 'r', 'c', and 'f' refer to rare, common and frequent category subsets.

3.2. Training Configuration

Experimental Details. We train our models for object detection and instance segmentation based on Detecron2[12], which is implemented in PyTorch. Unless specified, we use the ResNet backbone(pre-trained on ImageNet) with FPN[5]. Following the training procedure in [3], we resize the images so that the shorter side is 800 pixels. All baseline experiments are conducted on 8 GPUs with 2 images per GPU for 90K iterations, with a learning rate of 0.02 which is decreased by 10 at the 60K and 80K iteration. We use SGD with a weight decay of 0.0001 and momentum of 0.9. Scale jitter is applied for all experiments in default same with [3].

For the DisAlign, we freeze all network parameters and learn the magnitude and margin for extra 9K iterations with a learning rate of 0.02. Generalized re-weight is only used for fore-ground categories. Generalized re-weight scale ρ is set to 0.8 for all experiments.

3.3. Quantitative Results

We report the detailed results in Table.8 and Tab.9.

References

[1] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision(ECCV)*, 2018. 3, 4

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2009. 1

[3] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2019. 3, 5

[4] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. *International Conference on Learning Representations(ICLR)*, 2020. 1

[5] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2017. 5

[6] 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 *Proceedings of the European Conference on Computer Vision(ECCV)*, 2014. 5

[7] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2019. 1, 3

[8] Open MMLab. Mmsegmentation. https://github.com/open-mmlab/mmsegmentation, 2020. 3

[9] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems(NeurIPS)*, 2019. 1

[10] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence(TPAMI)*, 2017. 3, 4

[11] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2018. 1

[12] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 5

[13] Hang Zhang, Kristin Dana, Jianping Shi, Zhongyue Zhang, Xiaogang Wang, Ambrish Tyagi, and Amit Agrawal. Context encoding for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2018. 3

[14] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Zhi Zhang, Haibin Lin, Yue Sun, Tong He, Jonas Muller, R. Manmatha, Mu Li, and Alexander Smola. Resnest: Split-attention networks. *arXiv preprint*, 2020. 3

[15] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2017. 4