Boosting Long-tailed Object Detection via Step-wise Learning on Smooth-tail Data

Na Dong[1,2*] Yongqiang Zhang[2] Mingli Ding[2] Gim Hee Lee[1]
[1]Department of Computer Science, National University of Singapore
[2]School of Instrument Science and Engineering, Harbin Institute of Technology
{dongna1994, zhangyongqiang, dingml}@hit.edu.cn gimhee.lee@comp.nus.edu.sg

Abstract

Real-world data tends to follow a long-tailed distribution, where the class imbalance results in dominance of the head classes during training. In this paper, we propose a frustratingly simple but effective step-wise learning framework to gradually enhance the capability of the model in detecting all categories of long-tailed datasets. Specifically, we build smooth-tail data where the long-tailed distribution of categories decays smoothly to correct the bias towards head classes. We pre-train a model on the whole long-tailed data to preserve discriminability between all categories. We then fine-tune the class-agnostic modules of the pre-trained model on the head class dominant replay data to get a head class expert model with improved decision boundaries from all categories. Finally, we train a unified model on the tail class dominant replay data while transferring knowledge from the head class expert model to ensure accurate detection of all categories. Extensive experiments on long-tailed datasets LVIS v0.5 and LVIS v1.0 demonstrate the superior performance of our method, where we can improve the AP with ResNet-50 backbone from 27.0% to 30.3% AP, and especially for the rare categories from 15.5% to 24.9% AP. Our best model using ResNet-101 backbone can achieve 30.7% AP, which suppresses all existing detectors using the same backbone. Our source code is available at https://github.com/dongnana777/Long-tailed-object-detection.

1. Introduction

The success of deep learning are seen in many computer vision tasks including object detection. Many deep learning-based approaches [5, 29, 4, 17, 23, 20, 18, 1, 39] are proposed and have shown impressive performance in localizing and classifying objects of interest in 2D images. However, it is important for these deep learning-based approaches to be trained on balanced and representative datasets. Unfortunately, most real-world datasets always follow a long-tailed distribution, where the head classes have a significantly larger number of instances than the tail classes. Training on such imbalanced datasets often leads to bias towards head classes and significant performance degeneration of the tail classes due to the extremely scarce samples.

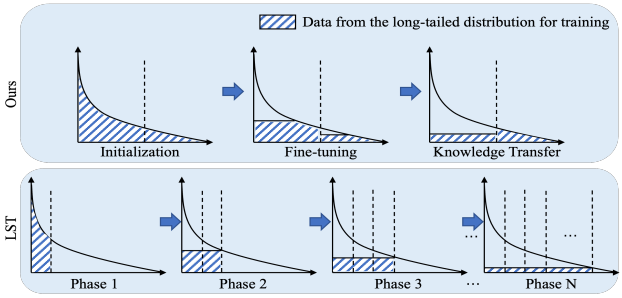

Figure 1. LST [10] is more susceptible to catastrophic forgetting due to their incremental learning scheme with numerous data splits. We alleviate the problem by building smooth-tail data that flattens long-tailed datasets and always maintains data from all categories.

To circumvent the long-tailed distribution problem of object detection task, many attempts exploit data re-sampling and loss re-weighting approaches. Data re-sampling methods [6, 31] re-balance the distribution of the instance numbers of each category. Loss re-weighting methods [28, 30, 15] adopt different re-weighting strategies to adjust the loss of different categories based on each category's statistics. As shown in Figure 2, Hu *et al*. [10] proposes LST which is a "divide & conquer" strategy that leverages class-incremental few-shot learning to solve the long-tailed distribution problem. The model is first trained with abundant labeled data of the head classes. The categories in the long-tailed training data is then sorted and divided according to the number of samples to get the corresponding subsets for incremental learning and merging of each part in N phases.

Despite the innovative adoption of class-incremental few-shot learning on the long-tailed distribution problem, we find that [10] catastrophically forgets the knowledge of the head classes and cannot sufficiently learn the tail classes in their incremental learning process. We postulate that this

*Work fully done while first author is a visiting PhD student at the National University of Singapore.

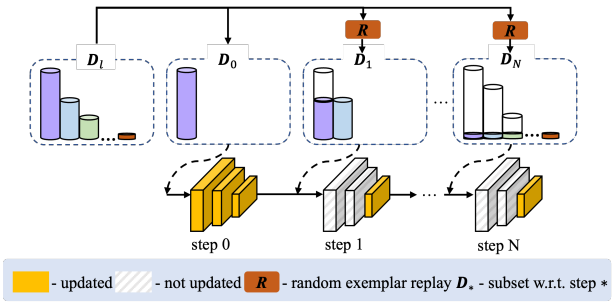

Figure 2. The incremental learning training strategy of [10] on numerous smaller and balanced data splits inevitably expedites catastrophic forgetting.

is attributed to three reasons: 1) Categories with high appearance similarity get divided into different parts due to the hard divisions. This leads to lower discriminability since these categories can only be trained together on the exemplar replay subsets. 2) There is an apparent discrepancy between the decision boundaries of the current model trained simultaneously on the exemplar replay subsets of the head and tail classes from the previous model trained solely on the head class subset. This discrepancy impedes the maintenance of the knowledge on the head classes and the learning of the tail classes. 3) The method divides the long-tailed dataset into numerous smaller balanced parts. However, this leads to more knowledge transfer steps and thus expediting catastrophic forgetting.

In this paper, we adopt a similar incremental few-shot learning approach to the long-tailed distribution object detection problem. To mitigate the above issues, we propose a simple but effective step-wise learning framework. We note that the main difference of long-tailed learning from class-incremental leaning is that the data of all categories can co-occur. In contrast to [10] that starts the training on only the head classes, we start the learning process from pre-training the model on the whole long-tailed dataset to better preserve the discriminative capability between the head and tail classes. In the subsequent steps, we keep the class-agnostic modules fixed and only update the class-specific modules of the pre-trained model trained on the whole long-tailed data. This circumvents the lack of training data in the tail end of the long-tailed data by preserving knowledge from the pre-trained model and limiting the network parameters that need to be updated.

To avoid severe catastrophic forgetting, we first divide all categories of long-tailed dataset into two parts: head classes with more than M images each category, and tail classes with less than M images each category. We then propose to build smooth-tail data: 1) a head class dominant data that contain a roughly balanced subset of the head classes minored with a roughly balanced subset of tail classes, and 2) a tail class dominant data in similar vein. We leverage the pre-trained model to select representative exemplars for the head class dominant and tail class dominant data. Subsequently, we fine-tune the pre-trained model on the head class dominant data to learn a head class expert model. Finally, we learn a unified model on the tail class dominant data while preserving knowledge of the head classes with the head class expert model. Knowledge distillation at feature level with a head class focused mask is adopt to facilitate the learning of tail classes from the head class expert model. In addition, knowledge distillation at classification head is also adopted, where object query features from the head class expert model are shared to the unified model to align the predictions between them.

Our contributions can be summarized as follows:

1. We propose to build smooth-tail data, *i.e.*, a head class dominant data and a tail class dominant data, to alleviate the extreme class imbalance of long-tail data and prevent catastrophic forgetting in our step-wise learning framework.
2. We design a novel step-wise learning framework that unifies fine-tuning and knowledge transfer for the long-tailed object detection task.
3. Our framework is frustratingly simple but effective. We achieve state-of-the-art performances on long-tailed datasets LVIS v0.5 and LVIS v1.0 in both the overall accuracy, and especially the impressive accuracy of the rare categories.

2. Related Works

General Object Detection. A large number of approaches have been proposed for object detection task, which can be briefly summarized into two different types based on their frameworks. Two-stage object detection methods such as R-CNN [5] apply a deep neural network to extract features from proposals generated by selective search [29]. Fast R-CNN [4] utilizes a differentiable RoI Pooling to improve the speed and performance. Faster R-CNN [24] introduces the Region Proposal Network to generate proposals. FPN [17] builds a top-down architecture with lateral connections to extract features across multiple layers. In contrast, one-stage object detection methods such as YOLO [23] directly perform object classification and bounding box regression on the feature maps. SSD [20] uses feature pyramid with different anchor sizes to cover the possible object scales. RetinaNet [18] proposes the focal loss to mitigate the imbalanced positive and negative examples. Recently, transformer-based object detection methods [1, 39] beyond the one-stage and two-stage methods have gained popularity, which achieve comparable or even better performance. They directly supervise bounding box predictions end-to-end with Hungarian bipartite matching. These object detection models require the training datasets to possess a roughly balanced category distribution, *e.g.* COCO dataset [19]. However, the

distribution of categories in the real-world scenarios is often long-tailed and most of these object detection models fail to maintain their performance. An extreme imbalance leads to low accuracy on tail classes.

Long-tailed Object Detection. Many existing works have been proposed to alleviate the challenge of long-tailed object detection. These works can be categorized into three categories. *Data re-sampling* is the most intuitive among all methods. Gupta *et al*. [6] proposes repeat factor sampling (RFS) to create a roughly balanced distribution by over-sampling data of tail classes based on the frequency of each category at image-level. Wang *et al*. [31] proposes a calibration framework to alleviate classification head bias with a bi-level class balanced sampling approach at instance-level. *Loss re-weighting* is another common approach. EQLv2 [28] adopts a gradient-guided mechanism to re-weight the loss contribution of each category. EFL [15] introduces a category-relevant modulating factor into focal loss to overcome the imbalance problem for one-stage object detectors. Wang *et al*. [30] proposes seesaw loss to re-balance gradients of positive and negative samples for each category, with two complementary factors. Wang *et al*. [32] proposes to understand the long-tailed distribution in a statistic-free perspective and present a adaptive class suppression loss. In addition to the above two common categories of methods, many works also approach the problem from different perspectives. AHRL [14] addresses long-tailed object detection from a metric learning perspective, which splits the whole feature space into hierarchical structure and eliminates the problem in a coarse-to-fine manner. Hu *et al*. [10] which mainly focuses on instance segmentation task proposes to alleviate long-tailed distribution problem in a class-incremental few-shot learning way.

Few-Shot Object Detection and Knowledge Transfer. Approaches of few-shot object detection can be categorized into meta-learning based [34, 11, 36, 38] and fine-tuning based methods [33, 35, 27]. There are two key differences between few-shot object detection and long-tailed object detection. On one hand, few-shot object detection merely focuses on the performance on few-shot categories, which is different from long-tailed object detection that aims at detecting all categories accurately. On the other hand, the datasets of few-shot object detection are comprised of base data which contains abundant training samples per category and novel data which contains a few training samples per category, which are quite different from long-tailed datasets.

Exemplar replay and knowledge distillation are two commonly used techniques to transfer knowledge across different models and remain performance of previous model. In exemplar replay based methods, the models strengthen memories learned in the past through replaying the past information periodically. They [22, 37, 2] usually keep a small number of exemplars per category to achieve this purpose. Knowledge distillation first proposed by Hinton *et al*. [8], where the knowledge of predicted distribution from the teacher model is distilled into the student model. Apart from the final prediction, other types of knowledge, like intermediate representations [26], can also be used to guide the learning of the student model.

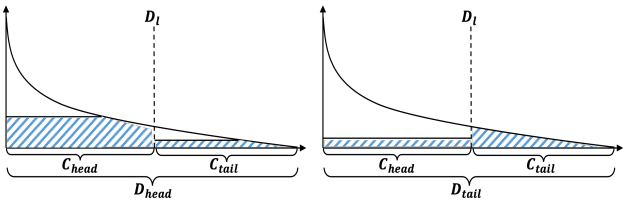

Figure 3. $\mathcal{D}_{head}$ contains a roughly balanced subset of $\mathcal{C}_{head}$ and a small roughly balanced subset of $\mathcal{C}_{tail}$. $\mathcal{D}_{tail}$ contains a roughly balanced subset of $\mathcal{C}_{tail}$ and a small balanced subset of $\mathcal{C}_{head}$.

Our proposed step-wise learning framework unifies fine-tuning and knowledge transfer techniques for the first time to alleviate the long-tailed distribution problem for object detection task, which can remain powerful on the head classes and better adapt to the tail classes.

3. Our Methodology

3.1. Dataset Pre-processing

As shown in Figure 3, given a long-tailed dataset $\mathcal{D}_l$ with $\mathcal{C}_l$ categories, we divide the entire set of categories into: the head classes $\mathcal{C}_{head}$ with each category containing $\geq M$ images, and the tail classes $\mathcal{C}_{tail}$ with each category containing $< M$ images. Furthermore, $\mathcal{C}_{head} \cup \mathcal{C}_{tail} = \mathcal{C}_l$ and $\mathcal{C}_{head} \cap \mathcal{C}_{tail} = \emptyset$. We then form $\mathcal{D}_{head}$ which is dominant with a roughly balanced subset of the head classes $\mathcal{C}_{head}$ and minored with a roughly balanced subset of the tail classes $\mathcal{C}_{tail}$. Similarly, we form $\mathcal{D}_{tail}$ which is dominant with a roughly balanced subset of the tail classes $\mathcal{C}_{tail}$ and minored with a balanced subset of the head classes $\mathcal{C}_{head}$.

Smooth-tail Data. We propose a confidence-guided exemplar replay scheme for the selection of representative and diverse exemplars in $\mathcal{D}_{head}$ and $\mathcal{D}_{tail}$. The number of exemplars is set to be significantly smaller than the original dataset. We propose to use the model pre-trained with the whole long-tailed data (*c.f*. next subsection) for the selection of the exemplars to ensure that the model trained on the few samples can also minimize the loss on the original dataset. Specifically, we save all instances and corresponding classification scores $\{I_j, S_j\}$ predicted by the pre-trained model for each category. We then sort the instances by the value of corresponding classification scores in a descending order. Finally, we select the top-scoring instances as representative exemplars for replay. Notably, only the annotations belonging to the selected instances are considered valid in the training process. Furthermore, the images in original

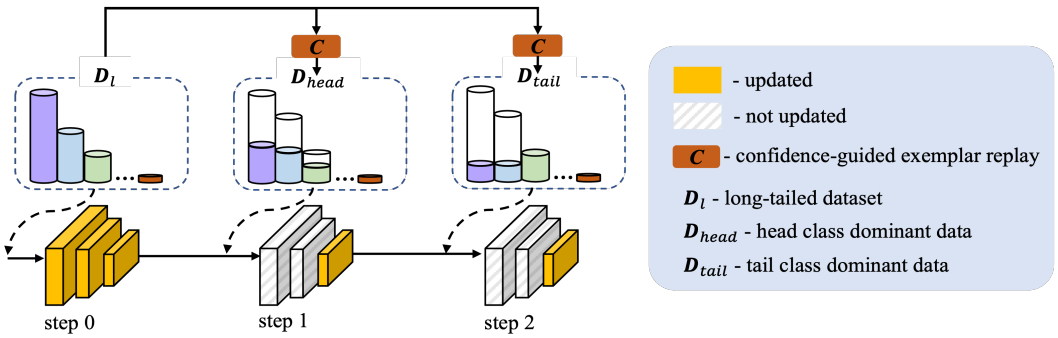

Figure 4. Overview of our step-wise learning framework. We first pre-train on the whole long-tailed training data $\mathcal{D}_l$, and then the class-specific modules are fine-tuned on $\mathcal{D}_{head}$. Finally, we train the model on $\mathcal{D}_{tail}$ while concurrently preserves knowledge from $\mathcal{D}_{head}$.

dataset are diverse in color, texture and size of region. The diversity of the exemplars ensures the same robustness and discrimination of the model as trained on original dataset, thus instances with classification scores greater than threshold 0.5 and are not in the same image are given the priority to be chosen as exemplars.

3.2. Step-wise Learning

We use the state-of-the-art Deformable DETR [39] as our backbone object detector. Given a long-tailed dataset $\mathcal{D}_l$ with $\mathcal{C}_l$ categories, we pre-train a model on all categories using the same loss functions as Deformable DETR. This pre-trained model serves to: 1) provide output classification confidences as instance selection cues for building the smooth-tail data; 2) learn discriminative representation and provide separation capability of all categories for subsequent fine-tuning on $\mathcal{D}_{head}$ and knowledge transfer on $\mathcal{D}_{tail}$.

As shown in Figure 4, we learn a head class expert model with fine-tuning, and adopt knowledge transfer from the head class expert model and the final model to unify the capability of detecting head and tail classes. As the learning proceeds, the model gradually approaches an optimal performance of all categories.

Fine-tuning on $\mathcal{D}_{head}$. We propose to only update the class-specific projection layer Φ_p and classification head Φ_{cls} with $\mathcal{D}_{head}$ while keeping the class-agnostic modules frozen. This is to impose a strong constraint on the previous representation and thus the discrimination representation does not shift severely in subsequent process. The model is fine-tuned with the standard Deformable DETR loss [39]. Note that $\mathcal{D}_{head}$ is dominant with a roughly balanced subset of $\mathcal{C}_{head}$ to alleviate class imbalance in the head classes, and minored with a roughly balanced subset of $\mathcal{C}_{tail}$ to make sure the decision boundary in the feature space has smaller gap compared to the final unified model in subsequent step.

Let the detection targets in $\mathcal{D}_{head}$ be denoted as $y = \{y_i\}_{i=1}^N = \{(c_i, b_i)\}_{i=1}^N$, where c_i and b_i are the object category and bounding box. Assume the N predictions for target category made by the model are $\hat{y} = \{\hat{y}_i\}_{i=1}^N = \{(\hat{p}(c_i), \hat{b}_i)\}_{i=1}^N$, where $\hat{p}(c_i)$ is probability of category c_i and $\hat{b}_i$ is the predicted bounding box. Following Deformable DETR, we compute the same matching cost between the prediction $\hat{y}_{\hat{\sigma}(i)}$ and the ground truth y_i using Hungarian algorithm [13], where $\hat{\sigma}(i)$ is the index computed by the optimal bipartite matching. The Hungarian loss for all matched pairs is thus defined as:

$$\mathcal{L}_{hg}(y, \hat{y}) = \sum_{i=1}^{N} [\mathcal{L}_{\text{cls}}(c_i, \hat{p}_{\hat{\sigma}(i)}(c_i)) + \mathbb{1}_{\{c_i \neq \varnothing\}} \mathcal{L}_{\text{box}}(b_i, \hat{b}_{\hat{\sigma}(i)})], \tag{1}$$

where $\mathcal{L}_{\text{cls}}$ is the sigmoid focal loss [18]. $\mathcal{L}_{\text{box}}$ is a linear combination of ℓ_1 loss and generalized IoU loss [25] with the same weight hyperparameters as Deformable DETR.

Knowledge Transfer on $\mathcal{D}_{tail}$. As shown in Figure 5, we keep the model fine-tuned on $\mathcal{D}_{head}$ fixed as the head class expert model. We also keep a unified model initialized with the parameters from the head class expert model, which we train on $\mathcal{D}_{tail}$ while preserving the knowledge from $\mathcal{D}_{head}$. Similar to the fine-tuning step, we also update only the class-specific projection layer Φ_p and classification head Φ_{cls} of the unified model while keeping the class-agnostic modules frozen. However, a naive constant updates of the projection layer and classification head on the tail classes can aggravate catastrophic forgetting of the head classes. We thus propose the use of exemplar replay and knowledge distillation to mitigate the catastrophic forgetting of the head classes.

As mentioned earlier, we keep a small but balanced replay exemplars of the head classes in $\mathcal{D}_{tail}$. The head class expert model is employed as an extra supervision signal to prevent the projection layer output features of the unified model from deviating too much from the output features of the head class expert model. On the other hand, we do not want the head class expert model to limit the learning process of the unified model on the tail classes. To this end, we introduce a head class focused binary mask $mask^{head}$ based on the ground-truth bounding boxes of the head classes to prevent negative influence on the tail class learning. Specifically, we set the value of the pixel on the feature map within the ground truth

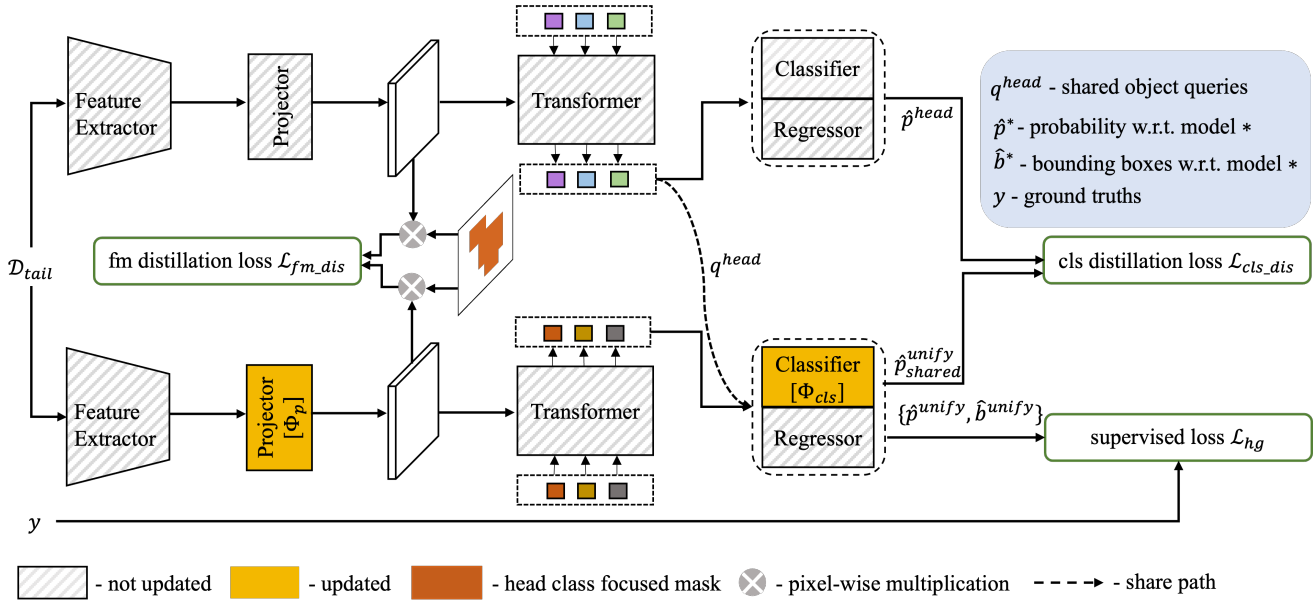

Figure 5. Overview of our proposed knowledge transfer. The framework consists of the fixed head class expert model (top branch) obtained from fine-tuning on $\mathcal{D}_{head}$ for knowledge transfer to the unified model (bottom branch) during training on $\mathcal{D}_{tail}$.

bounding boxes of head classes as 1, and the value of the pixel outside the ground truth bounding boxes as 0. The distillation loss on the features with the mask is written as:

$$\mathcal{L}_{fm\_dis} = \frac{1}{2N^{head}} \sum_{i=1}^{w} \sum_{j=1}^{h} \sum_{k=1}^{c} mask_{ij}^{head} \left\| f_{ijk}^{unify} - f_{ijk}^{head} \right\|^2, \quad (2)$$

where $N^{head} = \sum_{i=1}^{w} \sum_{j=1}^{h} mask_{ij}^{head}$. f^{head} and f^{unify} denote the features of the head class expert model and the unified model, respectively. w, h and c are the width, height and channels of the features.

Deformable DETR is built upon the transformer encoder-decoder architecture combined with a set-based Hungarian loss that forces unique predictions for each object via bipartite matching. Object queries extract features from the feature maps. Deformable DETR learns different spatial specialization for each object query, which indicates that different object queries focus on different position areas and box sizes. Since there is a mismatch in the object query features input into the classification head of the head class expert model and the unified model, the predicted classification outputs between the two models can be inevitably mismatched. To prevent the mismatch during knowledge distillation on the classification head, we first share the object query features q^{head} from the decoder output of the head class expert model to align the classification probability to the unified model. The classification outputs of the head class expert model and the unified model are compared in the distillation loss function given by:

$$\mathcal{L}_{cls\_dis} = \mathcal{L}_{kl\_div}(\log(\hat{p}^{unify}_{shared}(c_i)), \hat{p}^{head}(c_i)), \quad (3)$$

where we follow [8] in the definition of the KL-divergence loss $\mathcal{L}_{kl\_div}$ between the category probabilities of the head class expert model and the unified model. $\hat{p}^{unify}_{shared}(c_i)$ denotes the probability of category c_i with the shared object queries predicted by the unified model. $\hat{p}^{head}(c_i)$ denotes the probability of category c_i predicted by the head class expert model.

A Hungarian loss $\mathcal{L}_{hg}$ is also applied to the ground truth set y and the predictions $\hat{y}$ of the data of tail class dominant subset $\mathcal{D}_{tail}$. The overall loss $\mathcal{L}_{total}$ is given by:

$$\mathcal{L}_{total} = \mathcal{L}_{hg}(y, \hat{y}) + \lambda_{fm}\mathcal{L}_{fm\_dis} + \lambda_{cls}\mathcal{L}_{cls\_dis}. \quad (4)$$

λ_{fm} and λ_{cls} are hyperparameters to balance the loss terms.

4. Experiments

4.1. Experimental Settings

Datasets. To evaluate the performance of our proposed method, we conduct extensive experiments on the challenging LVIS v0.5 and LVIS v1.0 datasets. LVIS [6] is a large vocabulary dataset for long-tailed visual recognition. LVIS v0.5 contains 1230 categories, where 57k images in the $train$ set are used for training, and 5k images in the val set are used for validation. The latest version LVIS v1.0 contains 1203 categories, where 100k images with about 1.3M instances in the $train$ set are used for training, and 19.8k images in the val set are used for validation. All the categories are divided into three groups based on the number of images of each category that appear in the $train$ set: frequent (more than 100 images), common (10 to 100 images), and rare (less than 10 images). We report our results on the widely-used object detection metric AP^b across IoU threshold from 0.5

Method	Backbone	Dataset	AP^b	AP_r	AP_c	AP_f
LST [10]	ResNet-50	LVIS v0.5	22.6	-	-	-
DropLoss [9]			25.1	-	-	-
EQLv2 [28]			27.0	-	-	-
AHRL [14]			27.4	-	-	-
Our baseline			27.0	15.5	26.9	**31.6**
Ours			**30.3**	**24.9**	**31.5**	30.9
LST [10]	ResNet-101	LVIS v0.5	26.3	-	-	-
DropLoss [9]			26.8	-	-	-
EQLv2 [28]			28.1	-	-	-
AHRL [14]			29.3	-	-	-
Our baseline			27.0	14.6	27.3	**31.7**
Ours			**30.7**	**26.8**	**31.7**	31.1
BAGS [16]†	ResNet-50	LVIS v1.0	26.0	17.2	24.9	31.1
EQLv2 [28]†			25.5	16.4	23.9	31.2
Seesaw loss [30]†			26.4	17.5	25.3	31.5
AHRL [14]			26.4	-	-	-
EFL [15]†			27.5	20.2	26.1	32.4
Our baseline			25.1	11.9	23.1	**33.2**
Ours			**28.7**	**21.8**	**28.4**	32.0
BAGS [16]†	ResNet-101	LVIS v1.0	27.6	18.7	26.5	32.6
EQLv2 [28]†			26.9	18.2	25.4	32.4
Seesaw loss [30]†			27.8	18.7	27.0	32.8
AHRL [14]			28.7	-	-	-
EFL [15]†			29.2	23.5	27.4	**33.8**
Our baseline			26.3	14.4	24.8	33.2
Ours			**29.5**	**23.6**	**29.0**	32.6

Table 1. Comparisons with the state-of-the-art methods on LVIS v0.5 and LVIS v1.0 datasets. ResNet-50 and ResNet-101 are adopted as the backbones, respectively. † indicates results taken from [15].

Method	Framework	Backbone	Dataset	AP^b	AP_r	AP_c	AP_f
AHRL's baseline [14]	Mask R-CNN	ResNet-50	LVIS v0.5	26.7	-	-	-
AHRL [14]	Mask R-CNN			27.4	-	-	-
Our baseline	Deformable DETR			27.0	15.5	26.9	**31.6**
Ours	Deformable DETR			**30.3**	**24.9**	**31.5**	30.9
EFL's baseline [15]	RetinaNet	ResNet-50	LVIS v1.0	25.7	14.3	23.8	32.7
EFL [15]	RetinaNet			27.5	20.2	26.1	32.4
Our baseline	Deformable DETR			25.1	11.9	23.1	**33.2**
Ours	Deformable DETR			**28.7**	**21.8**	**28.4**	32.0

Table 2. Comparisons with the state-of-the-art methods and corresponding baselines.

to 0.95. Additionally, the boxes AP for frequent (AP_f), common (AP_c), and rare (AP_r) categories are also reported, respectively.

Implementation Details. We implement our method on Deformable DETR [39]. The ImageNet [3] pre-trained ResNet-50 and ResNet-101 [7] are adopted as the backbone. The training is carried out on 8 RTX 3090 GPUs with a batch size of 2 per GPU. We train our model using the AdamW [12, 21] optimizer with a weight decay of 1×10^{-4}. In the model pre-training step (step 0 of our framework), we train our model for 50 epochs with an initial learning rate of 2×10^{-4} and the learning rate is decayed at 40^{th} epoch by a factor of 0.1. In the model fine-tuning step (step 1 of our framework), the model is initialized from the pre-trained model. The parameters of the projection layer and classification head are updated while keeping the parameters of other modules frozen. We fine-tune the model for 1 epoch with a learning rate of 2×10^{-5}. In the knowledge transfer step (step 2 of our framework), the model is initialized from the fine-tuned model. The parameters of the projection layer and classification head are updated while keeping the other modules frozen. We train the model for 2 epochs with an initial learning rate of 2×10^{-4} and the learning rate is decayed at 1^{th} epoch by a factor of 0.1. λ_{fm} and λ_{cls} are set to

0.1 and 1, respectively. The hyperparameter M is set to 30.

4.2. Comparisons with the State-of-the-art Methods

To validate the effectiveness of our approach, we compare with state-of-the-art methods for long-tailed object detection on benchmark datasets LVIS v0.5 and LVIS v1.0. Our baseline is Deformable DETR [39] trained on long-tailed dataset $\mathcal{D}_l$ with the same loss functions as [39]. As shown in Table 1, our method achieves the best performance compared to all other existing methods. Specifically, our proposed method achieves 30.3% AP on LVIS v0.5 with ResNet-50 backbone. It improves the baseline by 3.3% AP, and even achieves 9.4% AP improvement on the rare categories. Our proposed method also outperforms the state-of-the-art AHRL [14] by 2.9% AP. With ResNet-101 as backbone, our approach still performs well on the baseline (+3.7% AP). Furthermore, our method outperforms the baseline by 3.6% AP with ResNet-50 backbone and 3.2% AP with ResNet-101 backbone on LVIS v1.0. The above results demonstrate that our method which unifies fine-tuning and knowledge transfer can effectively solve the severe class imbalance problem.

To eliminate the doubt that whether the gain is brought by different baselines, we present a more detailed comparison with the state-of-the-art methods on both the baselines and the final models. The results are present in Table 2. On LVIS v0.5, our method suppresses AHRL [14] by 2.9% AP with a slight advantage on baseline (AHRL's baseline: 26.7% AP vs Our baseline: 27.0% AP). On LVIS v1.0, while the performance of the baseline of EFL [15] is better than our baseline (EFL's baseline: 25.7% AP vs Our baseline: 25.1% AP), our method still outperforms EFL [15] by 1.2% AP and outperforms our baseline by 3.6% AP. Consequently, we can conclude that the improvements brought by our method benefit from our novel design instead of the different baseline.

4.3. Ablation Studies

FT	KT	AP^b	AP_r	AP_c	AP_f
		27.0	15.5	26.9	**31.6**
✓		29.7	19.4	31.4	**31.6**
	✓	29.4	23.2	29.8	31.3
✓	✓	**30.3**	**24.9**	**31.5**	30.9

Table 3. Ablation study of each component in our step-wise learning framework on the smooth-tail data. FT, KT indicate the fine-tuning and knowledge transfer, respectively.

Effectiveness of Each Component. There are two steps in our proposed step-wise learning framework, *i.e.*, fine-tuning on the head class dominant data and knowledge transfer on the tail class dominant data. We perform ablation study to demonstrate the effectiveness of each of them. As shown in Table 3, both the fine-tuning step and knowledge transfer step on the matched smooth-tail data play significant roles in step-wise learning framework.

For fine-tuning the model on the head class dominant data, it improves the performance of our baseline from 27.0% AP to 29.7% AP, while the performance improvement on rare categories is still limited (19.4% AP). We then examine the effectiveness of knowledge transfer. In this setting, we directly leverage the baseline as the extra supervision in knowledge transfer step instead of using the fine-tuned head class expert model as the extra supervision. Our method outperforms the baseline by 2.4% AP with significant improvement of the performance on the rare and common categories. However, the performance of the frequent categories experiences a slight drop.

Fine-tuning and knowledge transfer work collaboratively to achieve an improvement from 27.0% AP to 30.3% AP. Particularly, it achieves 24.9% AP for the rare categories, which outperforms the baseline by 9.4% AP and outperforms the fine-tuned head class expert by 5.5% AP. This indicates our proposed step-wise learning framework can sufficiently eliminate the class imbalance problem. However, our method experiences a further drop in the performance of the frequent categories after fine-tuning and knowledge transfer compared to using them separately (FT: 31.6% vs KT:31.3% vs FT&KT: 30.9% AP). We postulate that the drop in performance on the frequent categories might be due to insufficient representation of the frequent categories in our tail class dominant replay data during knowledge transfer. Similarly, the selection of a roughly balanced head classes for the head class dominant replay data might also result in under representation of the frequent categories. Consequently, catastrophic forgetting has a more detrimental effect on the frequent categories.

SOQ	$\mathcal{L}_{fm\_dis}$	$\mathcal{L}_{cls\_dis}$	AP^b	AP_r	AP_c	AP_f
✓	✓		29.4	24.9	30.6	29.8
✓		✓	29.7	**25.0**	30.8	30.3
	✓	✓	24.4	24.3	26.3	22.0
✓	✓	✓	**30.3**	24.9	**31.5**	**30.9**

Table 4. Ablation study of each component in our knowledge transfer. SOQ indicates the shared object queries.

Effectiveness of Each Component of Knowledge Transfer. We also demonstrate the effectiveness of each component of knowledge transfer. The results in Row 1 and Row 2 of Table 4 show that both knowledge distillation on features and knowledge distillation on classification output predictions play significant roles in knowledge transfer. It is worth noting that the performance decreases drastically when we do not share the object query features (from 30.3% AP to 24.4% AP), which can be attributed to the mismatch between the classification outputs of the head class expert model and the unified model.

Analysis of Divisions. The type of divisions on the long-tailed data plays an important role in our approach. We conduct extensive experiments to study the influence of

Division	AP^b	AP_r	AP_c	AP_f
$[1,10) \cup [10,-)$	30.1	24.7	31.1	30.8
$[1,30) \cup [30,-)$ (Ours)	**30.3**	**24.9**	31.5	30.9
$[1,50) \cup [50,-)$	30.2	24.1	**31.6**	31.0
$[1,100) \cup [100,-)$	30.1	23.9	31.3	**31.2**
$[1,10) \cup [10,100) \cup [100,-)$	29.8	23.7	31.0	30.7
$[1,10) \cup [10,30) \cup [30,100) \cup [100,-)$	29.3	24.8	30.4	29.8

Table 5. Ablation study of different type of divisions.

N_{ex} of $\mathcal{C}_{head}$	N_{ex} of $\mathcal{C}_{tail}$	AP^b	AP_r	AP_c	AP_f
100	30	30.1	**25.0**	31.3	30.6
200	30	**30.3**	24.9	**31.5**	30.9
300	30	30.0	24.6	31.0	**31.0**
200	10	30.2	24.7	31.3	**30.9**
200	30	**30.3**	24.9	**31.5**	**30.9**
200	50	30.2	25.0	31.3	30.8
200	100	30.2	**25.1**	**31.5**	30.8

Table 6. Ablation study of exemplar memory size of $\mathcal{D}_{head}$.

N_{ex} of $\mathcal{C}_{head}$	AP^b	AP_r	AP_c	AP_f
10	29.6	24.4	30.8	30.2
30	30.0	24.8	31.2	30.6
50	**30.3**	**24.9**	**31.5**	30.9
100	30.1	23.2	31.4	**31.3**

Table 7. Ablation study of exemplar memory size of $\mathcal{D}_{tail}$.

different type of divisions of the long-tailed dataset. As shown in Table 5, we can see that training the model with division $[1,30) \cup [30,-)$ achieves the best performance. All two-step divisions can outperform the performance of three-step or four-step divisions. We attribute this good performance to the fewer divisions, and the lower performance by the divisions $[1,10) \cup [10,100) \cup [100,-)$ and $[1,10) \cup [10,30) \cup [30,100) \cup [100,-)$ are caused by severe catastrophic forgetting from the increase in divisions. The performance of our two-step division $[1,30) \cup [30,-)$ also surpasses the other three two-step divisions, which clearly demonstrate the superiority of the division $[1,30) \cup [30,-)$ in adapting to the tail classes while maintaining the performance of the head classes.

Analysis of Exemplar Memory Size. We form $\mathcal{D}_{head}$ which is dominant with a roughly balanced subset of the head classes $\mathcal{C}_{head}$ and minored with a roughly balanced subset of the tail classes $\mathcal{C}_{tail}$. Similarly, we form $\mathcal{D}_{tail}$ which is dominant with a roughly balanced subset of the tail classes $\mathcal{C}_{tail}$ and minored with a balanced subset of the head classes $\mathcal{C}_{head}$. We denote N_{ex} as the number of instances per category. For $\mathcal{D}_{head}$ and $\mathcal{D}_{tail}$, we vary N_{ex} of the head classes $\mathcal{C}_{head}$ and the tail classes $\mathcal{C}_{tail}$ and report the results in Tables 6 and 7, respectively. We find that increasing N_{ex} of $\mathcal{C}_{head}$ helps maintain the performance of head classes. However, we also observe that increasing N_{ex} of $\mathcal{C}_{head}$ impedes the learning of tail classes and hurts the performance of tail classes. In addition, increasing N_{ex} of $\mathcal{C}_{tail}$ to large values does not significantly help the learning of the tail classes and slightly shows adverse affects on the performance of the head classes. By validation, we therefore store 200 instances per category of $\mathcal{C}_{head}$ and 30 instances per category of $\mathcal{C}_{tail}$ in $\mathcal{D}_{head}$. Similarly, in $\mathcal{D}_{tail}$, we store 50 instances per category of $\mathcal{C}_{head}$ and introduce all instances of $\mathcal{C}_{tail}$. This can eliminate the class imbalance between $\mathcal{C}_{head}$ and $\mathcal{C}_{tail}$ inside the exemplar sets and achieve a trade-off of the performance of all categories.

Method	AP^b	AP_r	AP_c	AP_f
Ours w/o step-wise RFS	29.6	19.0	**31.6**	**31.2**
Ours	**30.3**	**24.9**	31.5	30.9

Table 8. Ablation study of step-wise RFS.

Analysis of Step-wise RFS. Class imbalance still exists in the exemplar replay data for the head and tail classes due to the severe imbalance between categories of the long-tailed dataset, and thus hinders the learning of categories having fewer data. To narrow the imbalance in the exemplar replay data, we propose to adopt the repeat factor sampling (RFS) to over-sample the data from categories having fewer data. In our proposed step-wise learning framework, RFS is used in different ways in different steps and thus we terms it as step-wise RFS. In the fine-tuning step, for the head class dominant replay data, we over-sample the categories having fewer data among the dominant head classes. In the knowledge transfer step, we also over-sample the categories having few data among the dominant tail classes for the tail class dominant replay data. As shown in Table 8, the comparisons between our method using and without using step-wise RFS indicate that applying step-wise RFS does help alleviate the imbalance inside the subsets.

5. Conclusion

In this work, we propose a simple yet effective method that leverages incremental learning on the long-tailed distribution problem for the object detection task. We identify that a pre-trained model on the whole long-tailed dataset can achieve high discriminability in all categories for subsequent training steps. We propose to build the smooth-tail distributed data for calibrating the class imbalance in long-tailed datasets, and maintaining representative and diverse head and tail class exemplar replay data. We propose a novel step-wise learning framework that first fine-tune the pre-trained model on the head class dominant replay data to get the head class expert model. Subsequently, knowledge is transferred from the head class expert model to a unified model trained on the tail class dominant replay data. Our method brings large improvements with notable boost on the tail classes on different backbones and various long-tailed

datasets. Furthermore, our method achieves state-of-the-art performance on the challenging LVIS benchmarks for object detection task.

6. Acknowledgements

The first author is funded by a scholarship from the China Scholarship Council (CSC). This research is supported by the National Research Foundation, Singapore under its AI Singapore Programme (AISG Award No: AISG2-RP-2021-024), and the Tier 2 grant MOE-T2EP20120-0011 from the Singapore Ministry of Education.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020. 1, 2

[2] Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 233–248, 2018. 3

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 6

[4] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 1, 2

[5] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014. 1, 2

[6] 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*, pages 5356–5364, 2019. 1, 3, 5

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 6

[8] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 3, 5

[9] Ting-I Hsieh, Esther Robb, Hwann-Tzong Chen, and Jia-Bin Huang. Droploss for long-tail instance segmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 1549–1557, 2021. 6

[10] Xinting Hu, Yi Jiang, Kaihua Tang, Jingyuan Chen, Chunyan Miao, and Hanwang Zhang. Learning to segment the tail. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14045–14054, 2020. 1, 2, 3, 6

[11] Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. Few-shot object detection via feature reweighting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8420–8429, 2019. 3

[12] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[13] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 4

[14] Banghuai Li. Adaptive hierarchical representation learning for long-tailed object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2313–2322, 2022. 3, 6, 7

[15] Bo Li, Yongqiang Yao, Jingru Tan, Gang Zhang, Fengwei Yu, Jianwei Lu, and Ye Luo. Equalized focal loss for dense long-tailed object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6990–6999, 2022. 1, 3, 6, 7

[16] Yu Li, Tao Wang, Bingyi Kang, Sheng Tang, Chunfeng Wang, Jintao Li, and Jiashi Feng. Overcoming classifier imbalance for long-tail object detection with balanced group softmax. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10991–11000, 2020. 6

[17] Tsung Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 1, 2

[18] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 1, 2, 4

[19] 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 *European conference on computer vision*, pages 740–755. Springer, 2014. 2

[20] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*, 2016. 1, 2

[21] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[22] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 3

[23] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 1, 2

[24] Shaoqing Ren, Kaiming He, Ross Girshick, and Sun Jian. Faster r-cnn: Towards real-time object detection with region proposal networks. In *International Conference on Neural Information Processing Systems*, 2015. 2

[25] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 658–666, 2019. 4

[26] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014. 3

[27] Bo Sun, Banghuai Li, Shengcai Cai, Ye Yuan, and Chi Zhang. Fsce: Few-shot object detection via contrastive proposal encoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7352–7362, 2021. 3

[28] Jingru Tan, Xin Lu, Gang Zhang, Changqing Yin, and Quanquan Li. Equalization loss v2: A new gradient balance approach for long-tailed object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1685–1694, 2021. 1, 3, 6

[29] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013. 1, 2

[30] Jiaqi Wang, Wenwei Zhang, Yuhang Zang, Yuhang Cao, Jiangmiao Pang, Tao Gong, Kai Chen, Ziwei Liu, Chen Change Loy, and Dahua Lin. Seesaw loss for long-tailed instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9695–9704, 2021. 1, 3, 6

[31] Tao Wang, Yu Li, Bingyi Kang, Junnan Li, Junhao Liew, Sheng Tang, Steven Hoi, and Jiashi Feng. The devil is in classification: A simple framework for long-tail instance segmentation. In *European conference on computer vision*, pages 728–744. Springer, 2020. 1, 3

[32] Tong Wang, Yousong Zhu, Chaoyang Zhao, Wei Zeng, Jinqiao Wang, and Ming Tang. Adaptive class suppression loss for long-tail object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3103–3112, 2021. 3

[33] Xin Wang, Thomas E Huang, Trevor Darrell, Joseph E Gonzalez, and Fisher Yu. Frustratingly simple few-shot object detection. *arXiv preprint arXiv:2003.06957*, 2020. 3

[34] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Meta-learning to detect rare objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9925–9934, 2019. 3

[35] Jiaxi Wu, Songtao Liu, Di Huang, and Yunhong Wang. Multi-scale positive sample refinement for few-shot object detection. In *European Conference on Computer Vision*, pages 456–472. Springer, 2020. 3

[36] Xiongwei Wu, Doyen Sahoo, and Steven Hoi. Meta-rcnn: Meta learning for few-shot object detection. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1679–1687, 2020. 3

[37] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 374–382, 2019. 3

[38] Gongjie Zhang, Zhipeng Luo, Kaiwen Cui, and Shijian Lu. Meta-detr: Few-shot object detection via unified image-level meta-learning. *arXiv preprint arXiv:2103.11731*, 2, 2021. 3

[39] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 1, 2, 4, 6, 7