Generalized Few-Shot Object Detection without Forgetting

Zhibo Fan, Yuchen Ma, Zeming Li, Jian Sun
Megvii Technology

{fanzhibo, mayuchen, lizeming, sunjian}@megvii.com

Abstract

Recently few-shot object detection is widely adopted to deal with data-limited situations. While most previous works merely focus on the performance on few-shot categories, we claim that detecting all classes is crucial as test samples may contain any instances in realistic applications, which requires the few-shot detector to learn new concepts without forgetting. Through analysis on transfer learning based methods, some neglected but beneficial properties are utilized to design a simple yet effective few-shot detector, Retentive R-CNN. It consists of Bias-Balanced RPN to debias the pretrained RPN and Re-detector to find few-shot class objects without forgetting previous knowledge. Extensive experiments on few-shot detection benchmarks show that Retentive R-CNN significantly outperforms state-of-the-art methods on overall performance among all settings as it can achieve competitive results on few-shot classes and ***does not degrade the base class performance at all****. Our approach has demonstrated that the long desired never-forgetting learner is available in object detection.*

记忆力好的

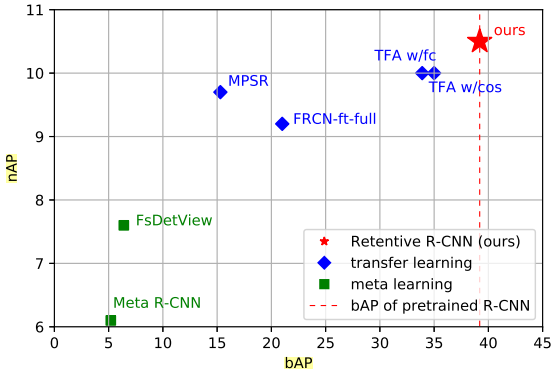

Figure 1. Performance of previous methods and ours on generalized few-shot objection on MS-COCO[25] under 10-shot settings, where base class AP and novel class AP are represented by x- and y-axis respectively. The red dashed line represents the base class AP of a base class detector. Our method does not degrade the base class AP while reaching state-of-the-art performance on novel categories.

1. Introduction

Computer vision community has seen significant progress by applying deep convolutional neural networks trained from a massive amount of data. However, sufficient training data is sometimes unavailable due to extensive human labor for annotation, especially for object detection, and the source data distribution may be long-tailed by nature such that certain object categories only contain limited examples. These circumstances raise the need to learn under a low-data regime effectively. Inspired by human's ability to learn new concepts rapidly from a handful of examples, few-shot learning[22, 19, 40, 37, 38, 30, 9, 8, 35, 3, 2, 42] is then proposed to mimic such generalization capability, with extensive research on image classification.

Several recent works[1, 17, 46, 18, 41, 44, 45, 14, 7, 6, 16, 43, 29] have attempted to apply few-shot learning techniques on instance-level tasks, such as object detection, where an extra localization task is included and more complicated visual contexts and features encountered, making few-shot object detection way more challenging. However, the majority focus merely on the performance of few-shot categories and ignore the catastrophic forgetting of base classes, which is not realistic. Unlike image classification, the capability to detect the joint domain of both classes at once is even crucial for object detection since samples at test time may contain instances of both classes, which requires the detector to be computationally efficient and learn new concepts without catastrophic forgetting. The problem of detecting objects of both classes is called Generalized Few-Shot Detection (G-FSD).

A popular stream of few-shot object detection[17, 46, 45, 14, 6] falls under the umbrella of meta-learning by leveraging external exemplars to do a visual search within the image. As their computational complexity is proportional to the number of categories, these methods become rather slow or even unavailable when tackling both sets of classes of a dataset. A promising alternative is transfer learning based

外部范本

approaches[1, 47, 41, 44], which can be trained incrementally to detect all classes in a single run. Wang *et al.*[41] share a similar interest in maintaining the overall performance on both classes and achieve competitive results by their two-stage finetuning approach (TFA), in which only the last layer of classification and box regression branch of RCNN[11, 10, 34] is finetuned while freezing backbone and RPN[34]. Nevertheless, there still exists a non-negligible base class performance gap with the pretrained model.

To diminish the gap, we first analyze the pretrained RCNN of TFA[41] and find advantageous but neglected properties: 1) pretrained base class detector does not predict many false positives on novel class instances despite their saliency 2) RPN is biased on its seen classes instead of being ideally class-agnostic, thus freezing it without exposure to new classes can be suboptimal. By utilizing these properties, we propose a simple yet effective transfer learning based method, Retentive R-CNN, to meet the demands of G-FSD to learn without forgetting and detect all categories efficiently. The name of Retentive R-CNN comes from its surprising ability to fully reserve the performance on base classes. Retentive R-CNN combines base and novel class detectors by Bias-Balanced RPN and Re-detector, introducing little extra cost. Bias-Balanced RPN can better adapt to novel class objects and remain powerful on the base class, thus provides better proposals for both training and inference. Re-detector utilizes a consistency loss to regularize the adaptation during finetuning and takes advantage of the base class detector's property to incrementally detect without forgetting. It is worth mentioning that our method does not degrade the base class performance at all while achieving competitive performance on novel classes as well, as shown in Figure1. Our contributions can be concluded as follows:

- We find properties of base class detectors neglected in few-shot detection literature, which can be utilized to improve both base and novel class performance for transfer learning based methods with little overhead.
- We propose a few-shot detector without forgetting, Retentive R-CNN, with Bias-Balanced RPN and Re-detector to assist novel class adaptation with base class knowledge and ensemble base and novel class detectors.
- Our method achieves state-of-the-art overall performance on the few-shot detection benchmark[41, 17] across all settings, with leading base class metrics and competitive novel class metrics.

2. Related Work

Few-Shot Learning. Previous few-shot learning literature mainly focuses on the task of image classification. Two popular approaches, metric learning[40, 37, 19, 38] and meta-learning[8, 35], have been widely adapted to avoid overfitting on the small data. Recent works[2, 42, 3] also demonstrate the effectiveness of a pretrained backbone as a strong feature extractor and outperform many previous methods. However, catastrophic forgetting[28, 27] on base classes may happen during finetuning. Gidaris *et al.*[9] stress that a good few-shot learning system should adapt to new tasks rapidly while maintaining the performance on previous knowledge without forgetting[28, 27], namely generalized few-shot learning, which is also the research interest of several other works[30, 31, 36, 21]. It is worth mentioning that such an ability is more critical for object detection since images may have instances of both sets of categories.

Object Detection. Modern object detection has seen tremendous progress by utilizing deep convolutional networks. One of the representative architectures is R-CNN[11, 10, 34, 12, 23], which generates object proposals upon the holistic image features, then classify and refine the proposals given the features within it. R-CNN is also the architecture mostly explored in the context of few-shot object detection and the one we extend for G-FSD in this paper. Impressive progress has also been made by single-stage methods[33, 26, 24] and recent anchor-free methods[20, 4, 39, 49, 32].

Few-Shot Object Detection. Exploration of few-shot object detection so far can be categorized into two streams: meta-learning[17, 46, 14, 29, 7, 6, 18, 45, 43, 16] based and transfer learning based[1, 41, 44, 47]. The majority of meta-learning stream predict detections conditioned on a set of support examples, which can be viewed as an exemplar-based visual search. For instance, Meta R-CNN[46] predicts upon ROI features reweighted by attentive vectors of each class, whose computational complexity grows linearly as the number of categories increases, making it hard to apply on large-scale datasets. On the contrary, transfer learning based methods can easily employ full class detection. Transfer learning methods thus far have explored various aspects: Chen *et al.*[1] apply regularizations during finetuning, Yang *et al.*[47] utilize a non-local structure to model global context, Wu *et al.*[44] augment training samples to mitigate scale bias due to limited data.

A handful of works share a similar focus on well detecting both classes: Juan-Manuel *et al.*[16] try to tackle it with a meta-learned CenterNet[4], though the performance is still limited with the linearly growing complexity issue; Wang *et al.*[41] propose TFA with a simple pretrain-finetune scheme for G-FSD. In TFA, adapting to novel classes with less degradation on base classes is achieved by two-stage finetuning: first finetune on novel classes, then use the weights of novel classes as initialization to finetune on both classes. The metrics on base classes are somewhat

reserved by this procedure and a slow learning schedule. Nevertheless, the performance drop in base classes still exists.

3. Approach

In this section, we start with the problem formulation of few-shot object detection. Next, we investigate the representative transfer learning based TFA[41] to reveal some neglected properties of the pretrained base detector. Then we describe our proposed model, which utilizes these properties, followed by training and inference details.

3.1. Problem Statement

Following previous literature[17, 41], we split the categories of a dataset into base classes $\mathcal{C}_b$ and novel classes $\mathcal{C}_n$, with $\mathcal{D}_b$ and $\mathcal{D}_n$ denoting the corresponding sub-datasets, respectively. $\mathcal{D}_b$ contains abundant annotations for training, while only a few data in $\mathcal{D}_n$ is available. Our objective is to learn a detection model $f(\cdot)$ for both $\mathcal{C}_b$ and $\mathcal{C}_n$ from the few novel class samples without forgetting the learned capability from the abundant base class samples.

Such an objective can be easily achieved by meta training a model to perform an exemplar-based visual search on $\mathcal{D}_b$, then directly deploy it without finetuning, as in one-shot detection literature [14, 29]. However, these methods do not perform as good as ordinary detection methods on $\mathcal{D}_b$ and require high time and space complexity. On the contrary, transfer learning based methods can efficiently deal with full-way detection and achieve competitive results on $\mathcal{C}_n$, as demonstrated in previous works[41, 44]. Thus we propose to tackle the problem of G-FSD in a transfer learning paradigm: first obtain a base model f^b by training on $\mathcal{D}_b$ and then obtain a novel model f^n via finetuning f^b on $\mathcal{D}_n$ (or a combination of $\mathcal{D}_b$'s subset and $\mathcal{D}_n$). However, the finetuning stage tends to degrade the base class performance due to the forgetting effect[28, 27] if it is finetuned on $\mathcal{D}_n$, or due to the sample limitation on $\mathcal{D}_b$ to balance class frequency if finetuned on both classes. Regarding this problem, a question is probably raised: is the degradation unavoidable?

Metric / Component	RPN	RCNN	drop
uAR@1000	34.1	8.5	**25.6**
AR@1000	61.1	54.7	6.4

Table 1. Recall between the output of RPN, the detector and the ground truths of unseen classes (uAR) and seen classes (AR).

Metric / RPN	pretrained	finetuned
AR@100	31.3	**34.2**
AR@1000	45.8	**48.0**

Table 2. Mean average recall between the output of RPN and ground truths of both base and novel classes.

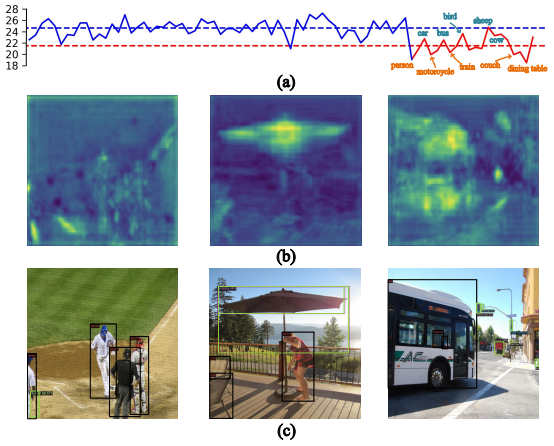

Figure 2. (a) L2-norms of ROI features extracted from a pretrained R-CNN on $\mathcal{D}^b$, sorted by class index. Blue and red represent seen and unseen classes, respectively, and dashed lines denote their average. Unseen classes with norms significantly higher or lower than average are annotated with class names in blue or red, respectively . (b) Backbone features' (FPN-P3) L2-norm map and (c) the corresponding detections of the base detector (green boxes) and ground truths of unseen classes (black boxes). The base detector has a strong ability to reject unseen classes.

3.2. Analysis on Transfer Learning based Few-Shot Object Detection

To answer this question without loss of generality, we analyze TFA[41]'s properties as a representative transfer learning model on few-shot detection tasks. TFA is first pretrained on $\mathcal{D}_b$ as ordinary R-CNN, then the last layers in classification and box regression heads are tuned on $\mathcal{D}_n$. The finetuned novel class heads' weights are concatenated with base class weights as the initialization for the final finetuning on a combined dataset consists of $\mathcal{D}_n$ and $\mathcal{D}_b$'s subset, where the number of samples per category is enforced to be identical. A slow and steady learning schedule is also applied during the final finetuning stage. Take 10-shot settings on MS-COCO for example, AP on $\mathcal{C}_b$ is better reserved than a pure finetuning baseline (31.8 to 35.0), though AP of the base class detector can achieve as high as 39.2.

Why cosine classifier works? Cosine classifier is commonly adapted in few-shot classification[9, 30] as cosine similarity bridged transfer learning and metric learning approach and generally performs well on base and novel class trade-off. The conclusion remains valid for TFA[41] as the base class performance is generally higher with a cosine classifier. We collect the ROI features from an R-CNN pretrained on $\mathcal{D}_b$ of MS-COCO[25] and compute the average pixel-wise L2-norm of $\mathcal{C}_b$ and $\mathcal{C}_n$. The results are shown in Figure2(a). A massive variation of norms between base classes and unseen novel classes can be easily observed. This may account for the effectiveness of cosine classifiers for being agnostic to feature norms. Also, the norms of unseen classes with closer relationship with seen classes are

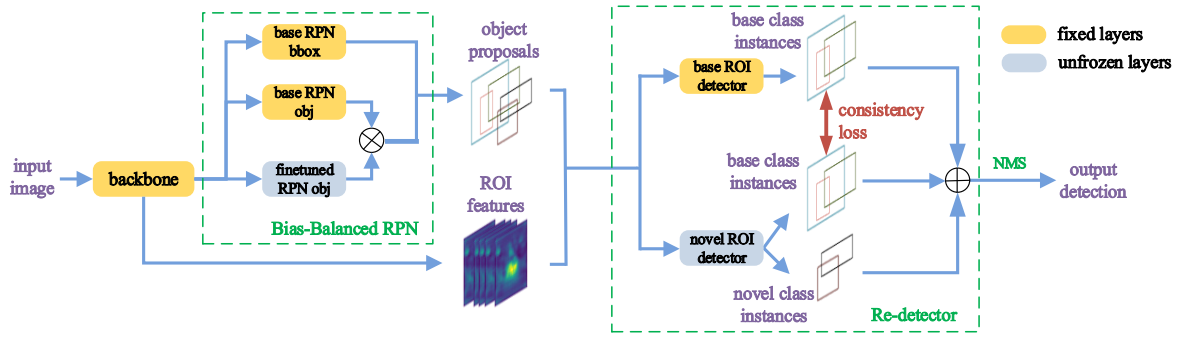

Figure 3. An overview of the proposed Retentive R-CNN. We implement Bias-Balanced RPN to debias the pretrained RPN and Re-detector to detect objects of both classes without forgetting, where a consistency loss is utilized to regularize finetuning. $\otimes$ represents ensembling operation, which is max in our implementation.

relatively higher (blue names annotated in Figure2(a)).

Does base detector find novel class salient objects? To a large extent, no. We hypothesize it is due to the features deactivated during training on $\mathcal{C}_b$ as indicated by low L2-norm, which will not produce a high confidence score with a dot-product classifier. We visualize the detection results and its feature norms on FPN[23] P3 in Figure2. The local features around the people in the first two images are obviously deactivated, although the object is of great saliency to humans. Features of the bus are somewhat activated in the third image of Figure2(b) (probably due to close relationship with trucks of $\mathcal{C}_b$), yet the detector is still able to recognize it as background. More results indicating this property are provided in the supplementary materials without cherry-picking. To quantitatively answer this question, the average recall between f^b's RPN proposals, final outputs, and ground truths of unseen class $\mathcal{C}_n$ (uAR) and seen classes (AR), is calculated in Table1. The drastic drop of uAR well demonstrates the ability of f^b to reject novel class objects. Thus we can utilize this property to reserve base class performance as f^b does not introduce many false positives on $\mathcal{C}_b$ when encountering novel class instances.

Is RPN class-agnostic? While most transfer learning[41, 44] and meta-learning[46, 45, 18] works treat RPN as class-agnostic and freeze it during finetuning, RPN is not ideally class-agnostic and biased on its seen categories. During training on $\mathcal{D}_b$, anchors of novel class instances are categorized into non-object due to lack of annotations, making RPN bias on training samples. We compare AR of *all* classes of a finetuned RPN on $\mathcal{C}_b \cup \mathcal{C}_n$ under 10-shot setting with pretrained RPN in Table2, where the apparent improvement validates our answer.

3.3. Retentive R-CNN

Our proposed model for G-FSD, Retentive R-CNN, consists of Bias-Balanced RPN and Re-detector to utilize the aforementioned properties of the base class detector f^b. The model architecture is illustrated in Figure3.

Re-detector. Re-detector consists of two detector heads, predicting detections of $\mathcal{C}_b$ and $\mathcal{C}_b \cup \mathcal{C}_n$ from object proposals in parallel, where one stream remains the same weight as in f^b to predict objects of $\mathcal{C}_b$ (denote as det^b) and the other holds the finetuned weights to detect objects of both $\mathcal{C}_n$ and $\mathcal{C}_b$ (denote as det^n). Detecting both classes can well alleviate the false positives due to inadequate data training, as shown in Section4.3. det^b utilize a fully-connected layer for classification and det^n use a cosine classifier to balance the variation of features in their norms. Similar to TFA, we finetune merely the last layers of classification and box regression head of det^n, which is capable of producing competitive results.

As f^b is trained from abundant data, we hope that det^n can inherit the reliable knowledge of f^b. Towards this end, we propose an auxiliary consistency loss to regularize det^n to score object proposals similar to det^b on the base class entries, which takes the form of KL-Divergence as in previous knowledge distillation works[15, 48]. For proposals of C_b, det^n is enforced to predict high confidence, and for proposals not belonging to $\mathcal{C}_b$, det^n mimics det^b with similarly low probabilities. Given the final probabilities p_c^b and p_c^n of class c predicted by det^b and det^n, the consistency loss is formalized as:

$$\mathcal{L}_{con} = \sum_{c \in \mathcal{C}_b} \tilde{p_c^n} log(\frac{\tilde{p_c^n}}{\tilde{p_c^b}}) \tag{1}$$

where $\tilde{p_i^n} = \frac{p_i^n}{\sum_{c \in \mathcal{C}_b} p_c^n}$ and the same for $\tilde{p_i^b}$. This is quite different from TK in LSTD[1] where the KL-Divergence is computed between the highest probabilities of $\mathcal{C}_b$ and $\mathcal{C}_n$. Note that p_i^n is the normalized marginal probability distribution over base classes after softmax over all class entries. The total loss of Re-detector during finetuning stage is

$$\mathcal{L}_{det} = \mathcal{L}_{cls}^n + \mathcal{L}_{box}^n + \lambda \mathcal{L}_{con} \tag{2}$$

where $\mathcal{L}_{cls}$ and $\mathcal{L}_{box}$ takes the same form as Faster R-CNN[34] and is computed on det^n only, and λ denotes the coefficient for the consistency loss.

Bias-Balanced RPN. R-CNN relies on RPN to generate object proposals as training samples for second stage classification and other subsequent processing. The quality of RPN proposals is especially crucial when the network is trained under low-data scenarios. As shown in Section3.2, a pretrained RPN may fail to catch novel class objects, further aggravating the scarcity of samples, while a finetuned RPN can alleviate this issue, thus providing better samples for second-stage modules to learn. We try to unfreeze different layers of RPN for finetuning and empirically, unfreeze the final layer that predicts objectness is sufficient to produce a noticeable improvement (results given in Setion4.3).

To retain performance on base classes, we propose Bias-Balanced RPN to integrate both pretrained RPN and the finetuned one. It ensembles the objectness prediction heads to raise $\mathcal{C}_b$ and $\mathcal{C}_n$ proposals properly. Given a feature map of size $H \times W$, base RPN predicts an objectness map $\mathcal{O}_b^{H\times W}$ and finetuned RPN predicts $\mathcal{O}_n^{H\times W}$, the final output objectness of Bias-Balanced RPN is defined as $\mathcal{O}^{H\times W} = \max(\mathcal{O}_b^{H\times W}, \mathcal{O}_n^{H\times W})$. Note that, during the finetuning stage, only the objectness of finetuned RPN is set unfrozen. Box regression and the convolution layer are shared across base RPN and finetuned RPN, as illustrated in Figure3. Theoretically, the max operation guarantees the RPN not to overlook proposals of previously learned classes catastrophically. With little computational overhead and extra weights, we believe Bias-Balanced RPN can serve as a general component for G-FSD. The full loss function of Retentive R-CNN during the finetuning stage is

$$\mathcal{L}_{ft} = \mathcal{L}_{obj}^{n} + \mathcal{L}_{det} \tag{3}$$

where $\mathcal{L}_{obj}^{n}$ is the binary cross-entropy loss on finetuned RPN's objectness layer.

Training. As a transfer learning based method, Retentive R-CNN is trained in two stages: pretraining on $\mathcal{D}_b$ and then finetune on the combined dataset of $\mathcal{D}_n$ and $\mathcal{D}_b$'s subset. As aforementioned, we only unfreeze three layers: objectness of the finetuned RPN, the last linear layers of classification and box regression of det^n. Thanks to the capability of retaining base class performance, we can apply a swifter learning schedule for finetuning, *e.g.*, 5000 iterations for 10-shot MS-COCO[25] compared to 160000 iterations of TFA[41].

Inference. Given the object proposals from Bias-Balanced RPN, the corresponding features are fed into the two heads of Re-detector in parallel. The set of predicted boxes of both heads are gathered into one for the final NMS procedure. As det^b is somehow more reliable as it learns from abundant data, we add a little bonus (0.1 in our implementation) for the scores predicted from det^b if they surpass the pre-NMS threshold, which could encourage the NMS procedure to take det^b's output when det^b and det^n find similar base class results. More details will be described in the supplementary material. As the backbone and feature transformation layers in Bias-Balanced RPN and Re-detector are shared among both detector heads, we can maintain the base class performance with little overhead compared to an ordinary R-CNN.

4. Experiments

4.1. Experimental Settings

We evaluate our method on the well-established few-shot detection benchmark[41, 17] based on MS-COCO[25] and Pascal VOC [5], following **the same class splits and data splits** in previous works[17, 41, 44] for a fair comparison. We report 5,10,30-shot results on MS-COCO and 1,2,3,5,10-shot results on 3 random splits of Pascal VOC. Towards the problem of G-FSD, **the overall performance of both classes is our major concern**. We reproduce Meta R-CNN[46] and FsDetView[45] using exactly the same samples for finetuning without hyperparameter changing (by running their official code) and denote the reproduced results with a * at the upper right corner. Results for ONCE[16], MetaDet[43] and FSRW[17] are reported from their original paper.

We use an ImageNet pretrained ResNet-101[13] with FPN[23] as the backbone. Pretraining on $\mathcal{D}_b$ is the same as in [41], then the finetuning layers are initialized by random. For all experiments, we set learning rate to 0.05 and λ to 0.1 to finetune until full convergence.

4.2. Comparison Experiments

We compared our results with both transfer learning[41, 44] and meta-learning based methods[16, 46, 17, 45]. Towards maintaining base class performance, one can quickly come up with an R-CNN model with N binary classifiers for detecting a dataset of N classes as binary classifiers are decoupled with each other. We also train such a model (denoted as FRCN-BCE) with binary cross-entropy loss for ROI classification as a strong baseline, using the same hyperparameters as Retentive R-CNN except for initializing the classifiers' bias as in RetinaNet[24].

Results on MS-COCO[25]. Table3 shows mean average precision over 0.5 to 0.95 IOU thresholds on all, base, and novel classes (AP, bAP, nAP) under different data settings. We outperform previous methods significantly on AP and bAP, as our method does not degrade on base classes at all. Meanwhile, we achieve competitive results on novel classes as well (state-of-the-art for 10-shot and on-par with state-of-the-art for 5- and 30-shot).

Towards the same objective to incrementally detect rare objects, ONCE[16] does not degrade the base class performance as well, yet its performance on both classes is limited. The very competitive TFA[41] models can gradually recover base class performance with samples increas-

Methods / Shots		5 shot			10 shot			30 shot		
		AP	bAP	nAP	AP	bAP	nAP	AP	bAP	nAP
Ours	Retentive R-CNN	**31.5**	**39.2**	**8.3**	**32.1**	**39.2**	**10.5**	**32.9**	**39.3**	**13.8**
Transfer Learning	FRCN-ft-full[41]	18.0	22.0	6.0	18.1	21.0	9.2	18.6	20.6	12.5
	FRCN-BCE	**29.1**	**36.8**	6.0	**29.2**	**36.8**	6.4	30.2	**36.8**	10.3
	TFA w/ fc[41]	27.5	33.9	**8.4**	27.9	33.9	**10.0**	29.7	35.1	13.4
	TFA w/ cos[41]	28.1	34.7	8.3	28.7	35.0	**10.0**	**30.3**	35.8	13.7
	MPSR[44]	-	-	-	15.3	17.1	9.7	17.1	18.1	**14.1**
Meta Learning	ONCE [16]	13.7	17.9	1.0	13.7	17.9	1.2	-	-	-
	Meta R-CNN*[46]	3.6	3.5	3.8	5.4	5.2	6.1	7.8	7.1	9.9
	FSRW[17]	-	-	-	-	-	5.6	-	-	9.1
	FsDetView*[45]	5.9	5.7	6.6	6.7	6.4	7.6	10.0	9.3	12.0

Table 3. Few-shot object detection results on MS-COCO under 5,10,30-shot settings, best viewed in color. AP, bAP, nAP represents mAP of MS-COCO for all classes, base classes, and novel classes, respectively. Best results and second-best are colored in red and blue, respectively, '-' means the result is not reported in the original paper. We outperform or on-par with all previous methods for each metric under these settings, with significant improvements on AP and bAP.

Methods / Shots		All Set 1					All Set 2					All Set 3				
		1	2	3	5	10	1	2	3	5	10	1	2	3	5	10
Ours	Retentive R-CNN	**71.3**	**72.3**	**72.1**	**74.0**	**74.6**	**66.8**	**68.4**	**70.2**	**70.7**	**71.5**	**69.0**	**70.9**	**72.3**	**73.9**	**74.1**
Transfer Learning	FRCN-ft-full[41]	55.4	57.1	56.8	60.1	60.9	50.1	53.7	53.6	55.9	55.5	58.5	59.1	58.7	61.8	60.8
	TFA w/ fc[41]	69.3	66.9	70.3	**73.4**	**73.2**	64.7	**66.3**	**67.7**	**68.3**	**68.7**	67.8	**68.9**	70.8	72.3	72.2
	TFA w/ cos[41]	**69.7**	**68.2**	**70.5**	**73.4**	72.8	**65.5**	65.0	**67.7**	68.0	68.6	**67.9**	68.6	**71.0**	**72.5**	**72.4**
	MPSR[44]	56.8	60.4	62.8	66.1	69.0	53.1	57.6	62.8	64.2	66.3	55.2	59.8	62.7	66.9	67.7
Meta Learning	Meta R-CNN*[46]	17.5	30.5	36.2	49.3	55.6	19.4	33.2	34.8	44.4	53.9	20.3	31.0	41.2	48.0	55.1
	FSRW[17]	53.5	50.2	55.3	56.0	59.5	55.1	54.2	55.2	57.5	58.9	54.2	53.5	54.7	58.6	57.6
	FsDetView*[45]	36.4	40.3	40.1	50.0	55.3	36.3	43.7	41.6	45.8	54.1	37.0	39.5	40.7	50.7	54.8

Table 4. Few-shot object detection results on Pascal VOC(07+12) **all classes (AP_{50})** under 1,2,3,5,10-shot settings, best viewed in color. Best results and second-best are colored in red and blue, respectively. Thanks to the non-forgetting ability, Retentive R-CNN consistently outperforms other methods w.r.t overall AP under all the data settings.

ing; however, the gap is still indispensable, *e.g.*, the bAP gap between 30-shot TFA w/cos[41] and the base model is as large as 3.4. As expected, FRCN-BCE can maintain base class performance from its pretrained model intrinsically, but the performance on both base and novel class is lower than an ordinary RCNN by a large margin. Given that Retentive R-CNN only adds little overhead with layers mostly shared, our method is a superior choice for G-FSD. Despite MPSR[44] slightly outperform our method with respect to nAP on 30-shot, the performance drop on base classes is significant, and thus it is not suitable for G-FSD. An even larger performance drop can be observed in Meta R-CNN[46] and FsDetView[45], probably because they predict the whole probability distribution of an ROI from features reweighted by a certain class-attentive vector. The vast performance drop is alleviated to some extent in FSRW[17] (see Table4), which only predicts the probability for the class of the reweighting vector.

Results on Pascal-VOC[5]. Table4 and Table5 show overall and novel class results on VOC benchmark respectively. Results of Meta R-CNN[46] from original paper are also included in Table5 as a reference. Note that the results are not directly comparable because samples used for finetuning are different, which can make a significant impact on the final metrics. We consistently outperform all methods on overall AP across all datasplits thanks to the non-forgetting property. As stated above, MPSR[44] and several other meta-learning methods[46, 45, 17] do not perform well on overall performance as the base class knowledge is forgotten during the finetuning stage.

Notice that performance on novel classes is not our primary concern, though, competitive results are achieved by Retentive R-CNN under most cases on VOC novel classes as shown in Table5. MPSR[44] made most of the best nAP records; however, non-negligible base class performance is sacrificed. Compared to the methods that better preserves base class performance, we outperform the current best TFA[41] in most cases, with approximative results under the rest.

4.3. Ablation Study and Visualization

Without loss of generality, we conduct ablation experiments on COCO benchmark under 10-shot scenario. All models are trained with the same hyperparameters unless otherwise stated.

Bias-Balanced RPN. To validate the effectiveness of our design, results on RPN recall and final detection precision for different classes of different RPN designs, including the

Methods / Shots		Novel Set 1					Novel Set 2					Novel Set 3				
		1	2	3	5	10	1	2	3	5	10	1	2	3	5	10
Ours	Retentive R-CNN	**42.4**	**45.8**	**45.9**	53.7	56.1	21.7	27.8	**35.2**	**37.0**	**40.3**	30.2	**37.6**	**43.0**	**49.7**	50.1
Transfer Learning	FRCN-ft-full[41]	15.2	20.3	29.0	25.5	28.7	13.4	20.6	28.6	32.4	38.8	19.6	20.8	28.7	42.2	42.1
	TFA w/ fc[41]	36.8	29.1	43.6	**55.7**	**57.0**	18.2	**29.0**	33.4	35.5	39.0	27.7	33.6	42.5	48.7	**50.2**
	TFA w/ cos[41]	39.8	36.1	44.7	**55.7**	56.0	**23.5**	26.9	34.1	35.1	39.1	30.8	34.8	42.8	49.5	49.8
	MPSR[44]	**42.8**	**43.6**	**48.4**	**55.3**	**61.2**	**29.8**	**28.1**	**41.6**	**43.2**	**47.0**	**35.9**	**40.0**	**43.7**	**48.9**	**51.3**
Meta Learning	Meta R-CNN[46]	19.9	25.5	35.0	45.7	51.5	10.4	19.4	29.6	34.8	45.4	14.3	18.2	27.5	41.2	48.1
	Meta R-CNN*[46]	16.8	20.1	20.3	38.2	43.7	7.7	12.0	14.9	21.9	31.1	9.2	13.9	26.2	29.2	36.2
	FSRW[17]	14.8	15.5	26.7	33.9	47.2	15.7	15.3	22.7	30.1	39.2	19.2	21.7	25.7	40.6	41.3
	MetaDet[43]	18.9	20.6	30.2	36.8	49.6	21.8	23.1	27.8	31.7	43.0	20.6	23.9	29.4	43.9	44.1
	FsDetView*[45]	25.4	20.4	37.4	36.1	42.3	22.9	21.7	22.6	25.6	29.2	**32.4**	19.0	29.8	33.2	39.8

Table 5. Few-shot object detection results on Pascal VOC(07+12) **novel classes (nAP$_{50}$)** under 1,2,3,5,10-shot settings, best viewed in color. Best results and second-best are colored in red and blue, respectively. Although nAP is **not** our primary concern, our method makes competitive results. MPSR[44] made most of the best nAP; however, base class metrics are largely sacrificed, as shown in Table4. We outperform TFA[41], the current best method paying fair attention to both classes, in most cases and on-par with it in the rest.

cls	bbox	AR	AP	bAP	nAP
max	**-**	**47.8**	**32.1**	39.2	**10.5**
-	-	45.6	32.0	**39.3**	10.1
arith-avg	-	47.1	32.0	**39.3**	10.3
geo-avg	-	33.5	30.5	37.4	9.6
max	unfreeze	47.7	32.0	**39.3**	10.4
max	arith-avg	47.7	32.0	39.2	10.4

Table 6. AR and AP results for base and novel classes among different RPN variants. Results of RPN components are ensembled by functions, including max, arithmetic average, and geometric average. '-' denotes for no ensembling, only base RPN is applied. The design option for current implementation is in bold.

$\mathcal{C}(f^n)$	$\mathcal{L}_{con}$	layers	cls	AP	bAP	nAP
all	**KLDiv**	**c+b**	**cos**	**32.1**	39.2	**10.5**
all	L1	c+b	cos	32.0	39.2	10.3
all	cos	c+b	cos	31.9	39.2	10.0
novel	-	c+b	cos	31.6	39.2	8.7
all	KLDiv	c+b+h	cos	31.9	39.2	9.8
all	KLDiv	c+b	fc	31.9	**39.3**	9.9

Table 7. Ablation results in Re-detector design. $\mathcal{C}(f^n)$ represents the classification domain of a novel detector. The column of layers denotes unfrozen layers in the second stage, c represents classification, b represents bbox and h represents linear layers in the box head. The design option for current implementation is in bold.

ensembling strategy for both RPN outputs and the choice of unfrozen layers during finetuning, are evaluated in Table6. Taking max as the ensembling strategy performs best, among other alternatives. Taking geometric average significantly degrades performance because any low objectness will produce a low final score. It can also be observed from the experiment that novel class AP is tightly related to AR of the RPN, while base class AP can remain stable with slightly inferior RPN AR, which validates one of our design philosophies to debias RPN thus improve novel class performance. Unfreezing box regression layers and ensembling does not make much difference. Thus the extra computation overhead is not necessary.

Re-detector. We study various design options in Re-detector, including the form of consistency loss (KL divergence, L1 difference, and negative cosine similarity between the normalized marginal probability distribution on base classes), layers in Re-detector to set unfrozen and classifier choice. As shown in Table7, our current design maximizes the overall performance. Surprisingly, unfreezing more layers even lowers the performance.

In addition, to validate the necessity of finetuning on both base and novel classes, we also implement a Re-detector where f^n only detects $\mathcal{C}_n$. It produces relatively low results, probably due to severer false positives as diverse objects in base classes are encountered during test time, but unseen during finetuning, and the model is trained to categorize these objects the same as those deactivated background features from only a handful of samples, which is undoubtedly challenging.

Inference time. We report average inference time per image on COCO 2014 test set by adding modules into Faster R-CNN in Table8. The inference time of Meta R-CNN[46] is also provided as a reference for representative meta-learning methods that demand exemplars for inference, which also introduces much lower extra computation than others, to the best of our knowledge. As most weights are set frozen and shared, Retentive R-CNN introduces little overhead during test time to realize few-shot detection without forgetting, especially compared to meta learned models requiring exemplars at test time.

FRCN	+BB-RPN	+ReDET	ours	Meta R-CNN
70.2	70.5	74.2	75.7	85.4

Table 8. Inference time in milliseconds on MS-COCO dataset. Meta R-CNN is reported in 10-shot from the original paper.

Visualization. We provide exemplary results obtained by Retentive R-CNN and TFA w/cos[41] in Figure4 for comparison under MS-COCO 10-shot setting. The non-forgetting property of our method can be observed from the last four images containing either crowded scenes or less

Figure 4. Visualization of Retentive R-CNN and TFA w/cos[41] results under MS-COCO 10-shot setting. Novel classes are bounded with purple boxes while base classes are bounded with green ones. Our method generally performs better on base classes and can detect novel class objects ignored by TFA in certain cases.

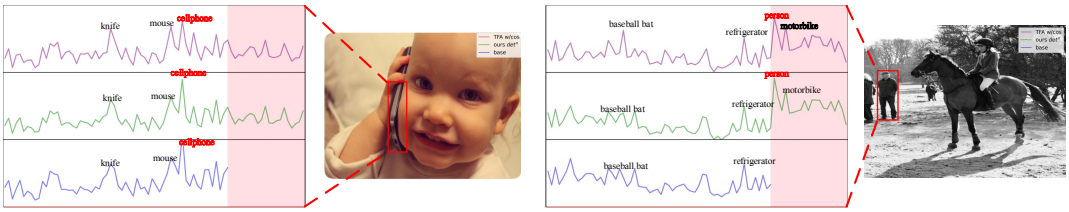

Figure 5. Visualization of classification logits of Retentive R-CNN det^n head, TFA w/cos[41] (finetuned from the same model), and the base model's classifiers. Regions colored pink represent novel class logits. The consistency loss regularizes our model to produce similar distribution as the base model on base classes, making detection on (a) base classes more reliable and (b) novel classes less confusing.

salient instances where TFA[41] tends to ignore some of these objects, *e.g.*, the inconspicuous baseball bat in the third image is ignored, and many well-learned objects are overseen in the fourth image by TFA[41]. We also perform better on novel classes under certain cases, as shown in the first two images.

We further investigate the role of consistency loss by comparing the classification distribution of our method and TFA w/cos[41] and a base detector. Specifically, we show two representative examples for the base class and novel class and visualize the logits of their classifiers for analysis. To make a fair comparison, both our method and TFA w/cos[41] are trained upon this same base detector. It can be easily observed that our method produces much more similar logits distribution as the base model on base classes rather than TFA w/cos[41]. Such property can better reserve base class performance, as shown in Figure5(a), where the base model and ours produce unimodal distribution with one strong peak. When it comes to novel classes, as shown in Figure5(b), base class distribution is suppressed, thus making a more confident response to novel classes.

5. Conclusion

In this paper, we have presented Retentive R-CNN to tackle the problem of G-FSD and proved that few-shot learning without forgetting is achievable in object detection. We analyze transfer learning based few-shot detection and find useful properties that are neglected by the community. Towards utilizing these properties, Retentive R-CNN is designed to combine base and novel detector simply and effectively, with Bias-Balanced RPN alleviating the bias of pre-trained RPN and Re-detector reliably finding objects of both base and novel classes. Experiments on well-established few-shot object detection benchmarks show that Retentive R-CNN does not degrade on the base class while remains competitive on novel classes, reaching state-of-the-art overall performance among all data settings. Ablation study validates the effectiveness of our design. Nevertheless, the huge performance gap between few-shot and general object detection on data-limited classes indicates that this task is arduous by nature, and we hope that this paper sheds light on works to further boost novel class metrics with little or no trade-off on base classes.

Acknowledgement

This research was partially supported by National Key R&D Program of China (No. 2017YFA0700800), and Beijing Academy of Artificial Intelligence (BAAI).

References

[1] Hao Chen, Yali Wang, Guoyou Wang, and Yu Qiao. Lstd: A low-shot transfer detector for object detection. In *AAAI Conference on Artificial Intelligence*, pages 2836–2843, 2018. 1, 2, 4

[2] Yinbo Chen, Xiaolong Wang, Zhuang Liu, Huijuan Xu, and Trevor Darrell. A new meta-baseline for few-shot learning. *arXiv preprint arXiv:2003.04390*, 2020. 1, 2

[3] Guneet S Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto. A baseline for few-shot image classification. *arXiv preprint arXiv:1909.02729*, 2019. 1, 2

[4] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. In *IEEE International Conference on Computer Vision*, pages 6569–6578, 2019. 2

[5] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. 5, 6

[6] Qi Fan, Wei Zhuo, and Yu-Wing Tai. Few-shot object detection with attention-rpn and multi-relation detector. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4012–4021, 2020. 1, 2

[7] Zhibo Fan, Jin-Gang Yu, Zhihao Liang, Jiarong Ou, Changxin Gao, Gui-Song Xia, and Yuanqing Li. Fgn: Fully guided network for few-shot instance segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9169–9178, 2020. 1, 2

[8] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, 2017. 1, 2

[9] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4367–4375, 2018. 1, 2, 3

[10] B. Ross Girshick. Fast r-cnn. In *IEEE International Conference on Computer Vision*, pages 1440–1448, 2015. 2

[11] B. Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014. 2

[12] Kaiming He, Georgia Gkioxari, Piotr Dollár, and B. Ross Girshick. Mask r-cnn. In *IEEE International Conference on Computer Vision*, pages 386–397, 2017. 2

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 5

[14] Ting-I Hsieh, Yi-Chen Lo, Hwann-Tzong Chen, and Tyng-Luh Liu. One-shot object detection with co-attention and co-excitation. In *Neural Information Processing Systems*, pages 2721–2730, 2019. 1, 2, 3

[15] Zhiting Hu, Zichao Yang, Ruslan Salakhutdinov, and Eric Xing. Deep neural networks with massive learned knowledge. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1670–1679, 2016. 4

[16] Perez-Rua Juan-Manuel, Zhu Xiatian, Hospedales Timothy, and Xiang Tao. Incremental few-shot object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 13843–13852, 2020. 1, 2, 5, 6

[17] Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. Few-shot object detection via feature reweighting. In *IEEE International Conference on Computer Vision*, pages 8419–8428, 2019. 1, 2, 3, 5, 6, 7

[18] Leonid Karlinsky, Joseph Shtok, Sivan Harary, Eli Schwartz, Amit Aides, Schmidt Rogério Feris, Raja Giryes, and M. Alexander Bronstein. Repmet - representative-based metric learning for classification and few-shot object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5197–5206, 2019. 1, 2, 4

[19] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2. Lille, 2015. 1, 2

[20] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *IEEE International Conference on Computer Vision*, pages 642–656, 2019. 2

[21] Aoxue Li, Tiange Luo, Tao Xiang, Weiran Huang, and Liwei Wang. Few-shot learning with global class representations. In *IEEE International Conference on Computer Vision*, pages 9715–9724, 2019. 2

[22] Fei-Fei Li, Fergus Rob, and Perona Pietro. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence*, 28(4):594–611, 2006. 1

[23] Tsung-Yi Lin, Piotr Dollár, B. Ross Girshick, Kaiming He, Bharath Hariharan, and J. Serge Belongie. Feature pyramid networks for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2, 4, 5

[24] Tsung-Yi Lin, Priya Goyal, B. Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision*, pages 318–327, 2017. 2, 5

[25] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and Lawrence C. Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pages 740–755, 2014. 1, 3, 5

[26] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, and Scott Reed. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*, 2016. 2

[27] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Neural Information Processing Systems*, pages 6467–6476, 2017. 2, 3

[28] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989. 2, 3

[29] Claudio Michaelis, Ivan Ustyuzhaninov, Matthias Bethge, and Alexander S. Ecker. One-shot instance segmentation. *arXiv preprint arXiv:1811.11507*, 2018. 1, 2, 3

[30] Hang Qi, Matthew Brown, and David G Lowe. Low-shot learning with imprinted weights. In *IEEE Conference*

on Computer Vision and Pattern Recognition, pages 5822–5830, 2018. 1, 2, 3

[31] Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan L Yuille. Few-shot image recognition by predicting parameters from activations. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 7229–7238, 2018. 2

[32] Han Qiu, Yuchen Ma, Zeming Li, Songtao Liu, and Jian Sun. Borderdet: Border feature for dense object detection. In *European Conference on Computer Vision*, pages 549–564. Springer, 2020. 2

[33] Joseph Redmon, Kumar Santosh Divvala, B. Ross Girshick, and ali farhadi. You only look once: Unified, real-time object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 2

[34] Shaoqing Ren, Kaiming He, B. Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1137–1149, 2017. 2, 4

[35] A. Andrei Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. In *International Conference on Learning Representations*, 2019. 1, 2

[36] Edgar Schonfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. Generalized zero-and few-shot learning via aligned variational autoencoders. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8247–8255, 2019. 2

[37] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Neural Information Processing Systems*, pages 4077–4087, 2017. 1, 2

[38] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1199–1208, 2018. 1, 2

[39] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *IEEE International Conference on Computer Vision*, pages 9627–9636, 2019. 2

[40] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Neural Information Processing Systems*, pages 3630–3638, 2016. 1, 2

[41] Xin Wang, Thomas Huang, Joseph Gonzalez, Trevor Darrell, and Fisher Yu. Frustratingly simple few-shot object detection. In *International Conference on Machine Learning*, 2020. 1, 2, 3, 4, 5, 6, 7, 8

[42] Yan Wang, Wei-Lun Chao, Kilian Q Weinberger, and Laurens van der Maaten. Simpleshot: Revisiting nearest-neighbor classification for few-shot learning. *arXiv preprint arXiv:1911.04623*, 2019. 1, 2

[43] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Meta-learning to detect rare objects. In *IEEE International Conference on Computer Vision*, pages 9924–9933, 2019. 1, 2, 5, 7

[44] 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*, 2020. 1, 2, 3, 4, 5, 6, 7

[45] Yang Xiao and Renaud Marlet. Few-shot object detection and viewpoint estimation for objects in the wild. In *European Conference on Computer Vision*, 2020. 1, 2, 4, 5, 6, 7

[46] Xiaopeng Yan, Ziliang Chen, Anni Xu, Xiaoxi Wang, Xiaodan Liang, and Liang Lin. Meta r-cnn - towards general solver for instance-level low-shot learning. In *IEEE International Conference on Computer Vision*, pages 9576–9585, 2019. 1, 2, 4, 5, 6, 7

[47] Ze Yang, Yali Wang, Xianyu Chen, Jianzhuang Liu, and Yu Qiao. Context-transformer: Tackling object confusion for few-shot detection. In *AAAI Conference on Artificial Intelligence*, pages 12653–12660, 2020. 2

[48] Ruichi Yu, Ang Li, Vlad I Morariu, and Larry S Davis. Visual relationship detection with internal and external linguistic knowledge distillation. In *Proceedings of the IEEE international conference on computer vision*, pages 1974–1982, 2017. 4

[49] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z. Li. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9756–9765, 2019. 2