Few-Shot Object Detection with Attention-RPN and Multi-Relation Detector

Qi Fan*
HKUST
qfanaa@cse.ust.hk

Wei Zhuo*
Tencent
wei.zhuowx@gmail.com

Chi-Keung Tang
HKUST
cktang@cse.ust.hk

Yu-Wing Tai
Tencent
yuwingtai@tencent.com

Abstract

Conventional methods for object detection typically require a substantial amount of training data and preparing such high-quality training data is very labor-intensive. In this paper, we propose a novel few-shot object detection network that aims at detecting objects of unseen categories with only a few annotated examples. Central to our method are our Attention-RPN, Multi-Relation Detector and Contrastive Training strategy, which exploit the similarity between the few shot support set and query set to detect novel objects while suppressing false detection in the background. To train our network, we contribute a new dataset that contains 1000 categories of various objects with high-quality annotations. To the best of our knowledge, this is one of the first datasets specifically designed for few-shot object detection. Once our few-shot network is trained, it can detect objects of unseen categories without further training or fine-tuning. Our method is general and has a wide range of potential applications. We produce a new state-of-the-art performance on different datasets in the few-shot setting. The dataset link is https://github.com/fanq15/Few-Shot-Object-Detection-Dataset.

1. Introduction

Existing object detection methods typically rely heavily on a huge amount of annotated data and require long training time. This has motivated the recent development of few-shot object detection. Few-shot learning is challenging given large variance of illumination, shape, texture, etc, in real-world objects. While significant research and progress have been made [1, 2, 3, 4, 5, 6, 7, 8], all of these methods focus on image classification rarely tapping into the problem of few-shot object detection, most probably because transferring from few-shot classification to few-shot object detection is a non-trivial task.

Central to object detection given only a few shots is how to localize an unseen object in a cluttered background, which in hindsight is a general problem of object localization from a few annotated examples in novel categories. Potential bounding boxes can easily miss unseen objects, or else many false detections in the background can be produced. We believe this is caused by the inappropriate low scores of good bounding boxes output from a region proposal network (RPN) making a novel object hard to be detected. This makes the few-shot object detection intrinsically different from few-shot classification. Recent works for few-shot object detection [9, 10, 11, 12] on the other hand all require fine-tuning and thus cannot be directly applied on novel categories.

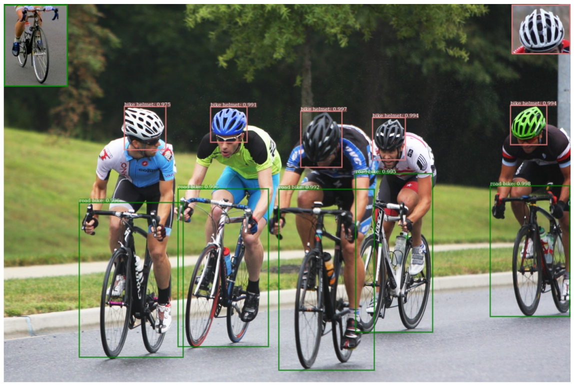

Figure 1. Given different objects as supports (top corners above), our approach can detect all objects in the same categories in the given query image.

In this paper, we address the problem of few-shot object detection: given a few support images of novel target object, our goal is to detect all foreground objects in the test set that belong to the target object category, as shown in Fig. 1. To this end, we propose two main contributions:

First, we propose a general few-shot object detection model that can be applied to detect novel objects without retraining and fine-tuning. With our carefully designed contrastive training strategy, attention module on RPN and detector, our method exploits matching relationship between object pairs in a weight-shared network at multiple network stages. This enables our model to perform online detection on objects of novel categories requiring *no* fine-training or further network adaptation. Experiments show that our model can benefit from the attention module at the early stage where the proposal quality is significantly enhanced, and from the multi-relation detector module at the

*Both authors contributed equally.

This research is supported in part by Tencent and the Research Grant Council of the Hong Kong SAR under grant no. 1620818.

later stage which suppresses and filters out false detection in the confusing background. Our model achieves new state-of-the-art performance on the ImageNet Detection dataset and MS COCO dataset in the few-shot setting.

The second contribution consists of a large well-annotated dataset with 1000 categories with only a few examples for each category. Overall, our method achieves significantly better performance by utilizing this dataset than existing large-scale datasets, *e.g*. COCO [13]. To the best of our knowledge, this is one of the first few-shot object detection datasets with an unprecedented number of object categories (1000). Using this dataset, our model achieves better performance on different datasets even without any fine-tuning.

2. Related Works

General Object Detection. Object detection is a classical problem in computer vision. In early years, object detection was usually formulated as a sliding window classification problem using handcrafted features [14, 15, 16]. With the rise of deep learning [17], CNN-based methods have become the dominant object detection solution. Most of the methods can be further divided into two general approaches: proposal-free detectors and proposal-based detectors. The first line of work follows a one-stage training strategy and does not explicitly generate proposal boxes [18, 19, 20, 21, 22]. On the other hand, the second line, pioneered by R-CNN [23], first extracts class-agnostic region proposals of the potential objects from a given image. These boxes are then further refined and classified into different categories by a specific module [24, 25, 26, 27]. An advantage of this strategy is that it can filter out many negative locations by the RPN module which facilitates the detector task next. For this sake, RPN-based methods usually perform better than proposal-free methods with state-of-the-art results [27] for the detection task. The methods mentioned above, however, work in an intensive supervision manner and are hard to extend to novel categories with only several examples.

Few-shot learning. Few-shot learning in a classical setting [28] is challenging for traditional machine learning algorithms to learn from just a few training examples. Earlier works attempted to learn a general prior [29, 30, 31, 32, 33], such as hand-designed strokes or parts which can be shared across categories. Some works [1, 34, 35, 36] focus on metric learning in manually designing a distance formulation among different categories. A more recent trend is to design a general agent/strategy that can guide supervised learning within each task; by accumulating knowledge the network can capture the structure variety across different tasks. This research direction is named meta-learning in general [2, 5, 37, 38, 39]. In this area, a siamese network was proposed in [37] that consists of twin networks sharing weights, where each network is respectively fed with a support image and a query. The distance between the query and its support is naturally learned by a logistic regression. This matching strategy captures inherent variety between support and query regardless of their categories. In the realm of matching framework, subsequent works [3, 4, 6, 8, 10, 40] had focused on enhancing feature embedding, where one direction is to build memory modules to capture global contexts among the supports. A number of works [41, 42] exploit local descriptors to reap additional knowledge from limited data. In [43, 44] the authors introduced Graph Neural Network (GNN) to model relationship between different categories. In [45] the given entire support set was traversed to identify task-relevant features and to make metric learning in high-dimensional space more effective. Other works, such as [2, 46], dedicate to learning a general agent to guide parameter optimization.

Until now, few-shot learning has not achieved groundbreaking progress, which has mostly focused on the classification task but rarely on other important computer vision tasks such as semantic segmentation [47, 48, 49], human motion prediction [50] and object detection [9]. In [51] unlabeled data was used and multiple modules were optimized alternately on images without box. However, the method may be misled by incorrect detection in weak supervision and requires re-training for a new category. In LSTD [9] the authors proposed a novel few-shot object detection framework that can transfer knowledge from one large dataset to another smaller dataset, by minimizing the gap of classifying posterior probability between the source domain and the target domain. This method, however, strongly depends on the source domain and is hard to extend to very different scenarios. Recently, several other works for few-shot detection [9, 10, 11, 12] have been proposed but they learn category-specific embeddings and require to be fine-tuned for novel categories.

Our work is motivated by the research line pioneered by the matching network [37]. We propose a general few-shot object detection network that learns the matching metric between image pairs based on the Faster R-CNN framework equipped with our novel attention RPN and multi-relation detector trained using our contrastive training strategy.

3. FSOD: A Highly-Diverse Few-Shot Object Detection Dataset

The key to few-shot learning lies in the generalization ability of the pertinent model when presented with novel categories. Thus, a high-diversity dataset with a large number of object categories is necessary for training a general model that can detect unseen objects and for performing convincing evaluation as well. However, existing datasets [13, 52, 53, 54, 55] contain very limited categories and they are not designed in the few-shot evaluation setting. Thus we build a new few-shot object detection dataset.

Dataset Construction. We build our dataset from existing large-scale object detection datasets for supervised learning *i.e*. [54, 56]. These datasets, however, cannot be used directly, due to 1) the label system of different datasets are

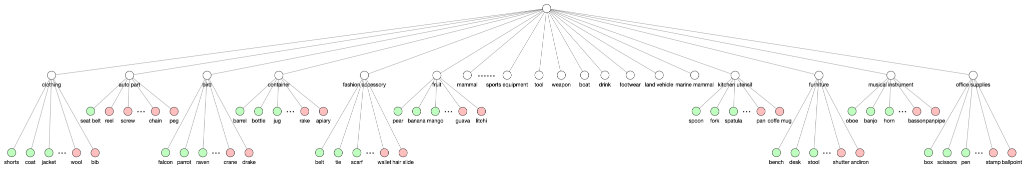

Figure 2. Dataset label tree. The ImageNet categories (red circles) are merged with Open Image categories (green circles) where the superclasses are adopted.

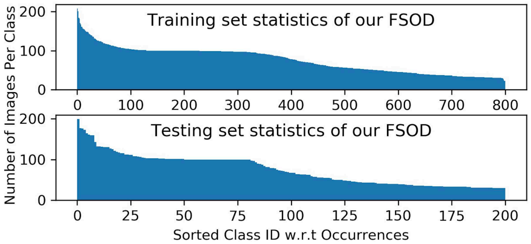

Figure 3. The dataset statistics of FSOD. The category image number are distributed almost averagely. Most classes (above 90%) has small or moderate amount of images (in [22, 108]), and the most frequent class still has no more than 208 images.

inconsistent where some objects with the same semantics are annotated with different words in the datasets; 2) large portion of the existing annotations are noisy due to inaccurate and missing labels, duplicate boxes, objects being too large; 3) their train/test split contains the same categories, while for the few-shot setting we want the train/test sets to contain different categories in order to evaluate its generality on unseen categories.

To start building the dataset, we first summarize a label system from [54, 56]. We merge the leaf labels in their original label trees, by grouping those in the same semantics (e.g., ice bear and polar bear) into one category, and removing semantics that do not belong to any leaf categories. Then, we remove the images with bad label quality and those with boxes of improper size. Specifically, removed images have boxes smaller than 0.05% of the image size which are usually in bad visual quality and unsuitable to serve as support examples. Next, we follow the few-shot learning setting to split our data into training set and test set without overlapping categories. We construct the training set with categories in MS COCO dataset [13] in case researchers prefer a pretraining stage. We then split the test set which contains 200 categories by choosing those with the largest distance with existing training categories, where the distance is the shortest path that connects the meaning of two phrases in the is-a taxonomy [57]. The remaining categories are merged into the training set that in total contains 800 categories. In all, we construct a dataset of 1000 categories with unambiguous category split for training and testing, where 531 categories are from ImageNet dataset [56] and 469 from Open Image dataset [54].

Dataset Analysis. Our dataset is specifically designed for few-shot learning and for evaluating the generality of a model on novel categories, which contains 1000 categories with 800/200 split for training and test set respectively, around 66,000 images and 182,000 bounding boxes in total. Detailed statistics are shown in Table 1 and Fig. 3. Our dataset has the following properties:

	Train	Test
No. Class	800	200
No. Image	52350	14152
No. Box	147489	35102
Avg No. Box / Img	2.82	2.48
Min No. Img / Cls	22	30
Max No. Img / Cls	208	199
Avg No. Img / Cls	75.65	74.31
Box Size	[6, 6828]	[13, 4605]
Box Area Ratio	[0.0009, 1]	[0.0009, 1]
Box W/H Ratio	[0.0216, 89]	[0.0199, 51.5]

Table 1. Dataset Summary. Our dataset is diverse with large variance in box size and aspect ratio.

High diversity in categories: Our dataset contains 83 parent semantics, such as mammal, clothing, weapon, etc, which are further split to 1000 leaf categories. Our label tree is shown in Fig. 2. Due to our strict dataset split, our train/test sets contain images of very different semantic categories thus presenting challenges to models to be evaluated.

Challenging setting: Our dataset contains objects with large variance on box size and aspect ratios, consisting of 26.5% images with no less than three objects in the test set. Our test set contains a large number of boxes of categories *not* included in our label system, thus presenting great challenges for a few-shot model.

Although our dataset has a large number of categories, the number of training images and boxes are much less than other large-scale benchmark datasets such as MS COCO dataset, which contains 123,287 images and around 886,000 bounding boxes. Our dataset is designed to be compact while effective for few-shot learning.

4. Our Methodology

In this section, we first define our task of few-shot detection, followed by a detailed description of our novel few-shot object detection network.

4.1. Problem Definition

Given a support image s_c with a close-up of the target object and a query image q_c which potentially contains objects of the support category c, the task is to find all the target objects belonging to the support category in the query and label them with tight bounding boxes. If the support set contains N categories and K examples for each category,

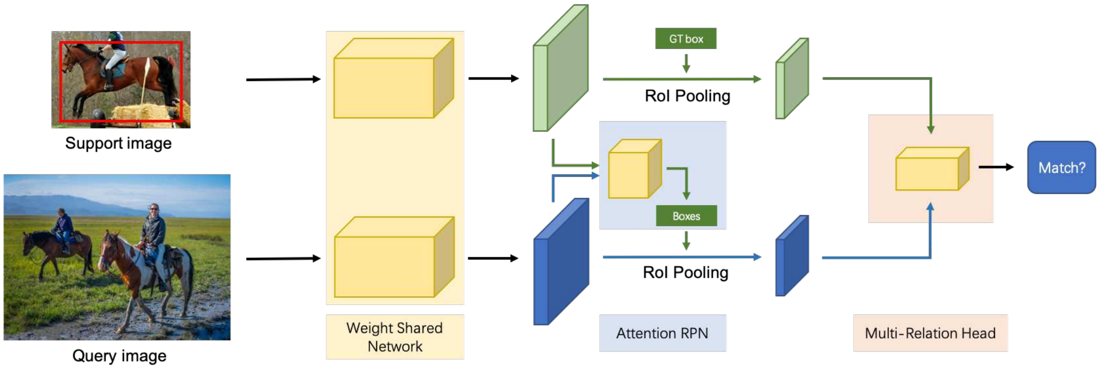

Figure 4. Network architecture. The query image and support image are processed by the weight-shared network. The attention RPN module filters out object proposals in other categories by focusing on the given support category. The multi-relation detector then matches the query proposals and the support object. For the N-way training, we extend the network by adding $N-1$ support branches where each branch has its own attention RPN and multi-relation detector with the query image. For K-shot training, we obtain all the support feature through the weight-shared network and use the average feature across all the supports belonging to the same category as its support feature.

the problem is dubbed N-way K-shot detection.

4.2. Deep Attentioned Few-Shot Detection

We propose a novel attention network that learns a general matching relationship between the support set and queries on both the RPN module and the detector. Fig. 4 shows the overall architecture of our network. Specifically, we build a weight-shared framework that consists of multiple branches, where one branch is for the query set and the others are for the support set (for simplicity, we only show one support branch in the figure). The query branch of the weight-shared framework is a Faster R-CNN network, which contains RPN and detector. We utilize this framework to train the matching relationship between support and query features, in order to make the network learn general knowledge among the same categories. Based on the framework, we introduce a novel attention RPN and detector with multi-relation modules to produce an accurate parsing between support and potential boxes in the query.

4.2.1 Attention-Based Region Proposal Network

In few-shot object detection, RPN is useful in producing potentially relevant boxes for facilitating the following task of detection. Specifically, the RPN should not only distinguish between objects and non-objects but also filter out negative objects not belonging to the support category. However, without any support image information, the RPN will be aimlessly active in every potential object with high objectness score even though they do not belong to the support category, thus burdening the subsequent classification task of the detector with a large number of irrelevant objects. To address this problem, we propose the attention RPN (Fig. 5) which uses support information to enable filtering out most background boxes and those in non-matching categories. Thus a smaller and more precise set of candidate proposals is generated with high potential containing target objects.

We introduce support information to RPN through the attention mechanism to guide the RPN to produce relevant proposals while suppressing proposals in other categories.

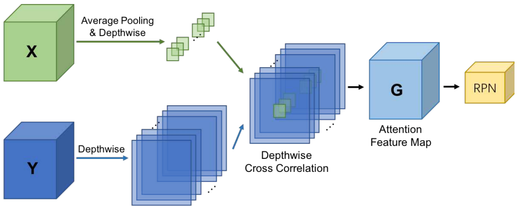

Figure 5. Attention RPN. The support feature is average pooled to a $1 \times 1 \times C$ vector. Then the depth-wise cross correlation with the query feature is computed whose output is used as attention feature to be fed into RPN for generating proposals.

Specifically, we compute the similarity between the feature map of support and that of the query in a depth-wise manner. The similarity map then is utilized to build the proposal generation. In particular, we denote the support features as $X \in t^{S\times S\times C}$ and feature map of the query as $Y \in t^{H\times W\times C}$, the similarity is defined as

$$\mathbf{G}_{h,w,c} = \sum_{i,j} X_{i,j,c} \cdot Y_{h+i-1,w+j-1,c}, \quad i,j \in \{1, ..., S\}$$

where $\mathbf{G}$ is the resultant attention feature map. Here the support features X is used as the kernel to slide on the query feature map [58, 59] in a depth-wise cross correlation manner [60]. In our work, we adopt the features of top layers to the RPN model, *i.e.* the res4_6 in ResNet50. We find that a kernel size of $S = 1$ performs well in our case. This fact is consistent with [25] that global feature can provide a good object prior for objectness classification. In our case, the kernel is calculated by averaging on the support feature map. The attention map is processed by a 3×3 convolution followed by the objectiveness classification layer and box regression layer. The attention RPN with loss L_{rpn} is trained jointly with the network as in [25].

4.2.2 Multi-Relation Detector

In an R-CNN framework, an RPN module will be followed by a detector whose important role is re-scoring proposals

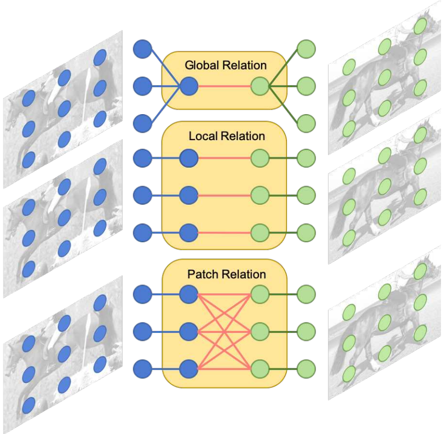

Figure 6. Multi-Relation Detector. Different relation heads model different relationships between the query and support image. The global relation head uses global representation to match images; local relation head captures pixel-to-pixel matching relationship; patch relation head models one-to-many pixel relationship.

and class recognition. Therefore, we want a detector to have a strong discriminative ability to distinguish different categories. To this end, we propose a novel multi-relation detector to effectively measure the similarity between proposal boxes from the query and the support objects, see Fig. 6. The detector includes three attention modules, which are respectively the **global-relation head** to learn a deep embedding for global matching, the **local-correlation head** to learn the pixel-wise and depth-wise cross correlation between support and query proposals and the **patch-relation head** to learn a deep non-linear metric for patch matching. We experimentally show that the three matching modules can complement each other to produce higher performance. Refer to the supplemental material for implementation details of the three heads.

Which relation heads do we need? We follow the N-way K-shot evaluation protocol proposed in RepMet [61] to evaluate our relation heads and other components. Table 2 shows the ablation study of our proposed multi-relation detector under the naive 1-way 1-shot training strategy and 5-way 5-shot evaluation on the FSOD dataset. We use the same evaluation setting hereafter for all ablation studies on the FSOD dataset. For individual heads, the local-relation head performs best on both AP_{50} and AP_{75} evaluations. Surprisingly, the patch-relation head performs worse than other relation heads, although it models more complicated relationship between images. We believe that the complicated relation head makes the model difficult to learn. When combining any two types of relation head, we obtain better performance than that of individual head. By combining all relation heads, we obtain the full multi-relation detector and achieve the best performance, showing that the three proposed relation heads are complementary to each other for better differentiation of targets from non-matching objects. All the following experiments thus adopt the full multi-relation detector.

Global R	Local R	Patch R	AP_{50}	AP_{75}
✓			47.7	34.0
	✓		50.5	35.9
		✓	45.1	32.8
✓		✓	49.6	35.9
	✓	✓	53.8	38.0
✓	✓		54.6	38.9
✓	✓	✓	**55.0**	**39.1**

Table 2. Experimental results for different relation head combinations in the 1-way 1-shot training strategy.

4.3. Two-way Contrastive Training Strategy

A naive training strategy is matching the same category objects by constructing a training pair (q_c, s_c) where the query image q_c and support image s_c are both in the same c-th category object. However a good model should not only match the same category objects but also distinguish different categories. For this reason, we propose a novel 2-way contrastive training strategy.

According to the different matching results in Fig. 7, we propose the 2-way contrastive training to match the same category while distinguishing different categories. We randomly choose one query image q_c, one support image s_c containing the same c-th category object and one other support image s_n containing a different n-th category object, to construct the training triplet (q_c, s_c, s_n), where $c \neq n$. In the training triplet, only the c-th category objects in the query image are labeled as foreground while all other objects are treated as background.

During training, the model learns to match every proposal generated by the attention RPN in the query image with the object in the support image. Thus the model learns to not only match the same category objects between (q_c, s_c) but also distinguish objects in different categories between (q_c, s_n). However, there are a massive amount of background proposals which usually dominate the training, especially with negative support images. For this reason, we balance the ratio of these matching pairs between query proposals and supports. We keep the ratio as 1:2:1 for the foreground proposal and positive support pairs (p_f, s_p), background proposal and positive support pairs (p_b, s_p), and proposal (foreground or background) and negative support pairs (p, s_n). We pick all N (p_f, s_p) pairs and select top $2N$ (p_b, s_p) pairs and top N (p, s_n) pairs respectively according to their matching scores and calculate the matching loss on the selected pairs. During training, we use the multi-task loss on each sampled proposal as $L = L_{matching} + L_{box}$ with the bounding-box loss L_{box} as defined in [24] and the matching loss being the binary cross-entropy.

Which training strategy is better? Refer to Table 3. We train our model with the 2-way 1-shot contrastive training strategy and obtain 7.9% AP_{50} improvement compared with the naive 1-way 1-shot training strategy, which indicates the importance in learning how to distinguish different categories during training. With 5-shot training, we achieve further improvement which was also verified in [1] that few-

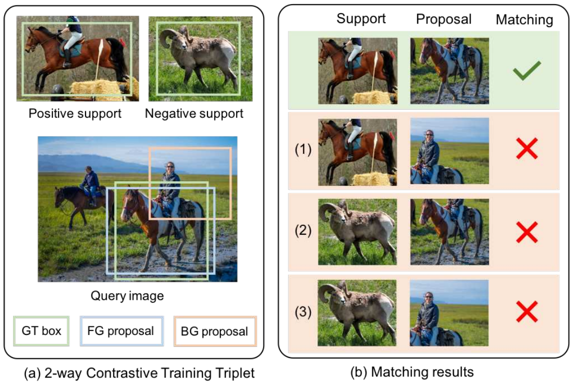

Figure 7. The 2-way contrastive training triplet and different matching results. Only the positive support has the same category with the target ground truth in the query image. The matching pair consists of the positive support and foreground proposal, and the non-matching pair has three categories: (1) positive support and background proposal, (2) negative support and foreground proposal and (3) negative support and background proposal.

shot training is beneficial to few-shot testing. It is straightforward to extend our 2-way training strategy to multi-way training strategy. However, from Table 3, the 5-way training strategy does not produce better performance than the 2-way training strategy. We believe that only one negative support category suffices in training the model for distinguishing different categories. Our full model thus adopts the 2-way 5-shot contrastive training strategy.

Which RPN is better? We evaluate our attention RPN on different evaluation metrics. To evaluate the proposal quality, we first evaluate the recall on top 100 proposals over 0.5 IoU threshold of the regular RPN and our proposed attention RPN. Our attention RPN exhibits better recall performance than the regular RPN (0.9130 *vs*. 0.8804). We then evaluate the average best overlap ratio (ABO [62]) across ground truth boxes for these two RPNs. The ABO of attention RPN is 0.7282 while the same metric of regular RPN is 0.7127. These results indicate that the attention RPN can generate more high-quality proposals.

Table 3 further compares models with attention RPN and those with the regular RPN in different training strategies. The model with attention RPN consistently performs better than the regular RPN on both AP_{50} and AP_{75} evaluation. The attention RPN produces 0.9%/2.0% gain in the 1-way 1-shot training strategy and 2.0%/2.1% gain in the 2-way 5-shot training strategy on the AP_{50}/AP_{75} evaluation. These results confirm that our attention RPN generates better proposals and benefits the final detection prediction. The attention RPN is thus adopted in our full model.

5. Experiments

In the experiments, we compare our approach with state-of-the-art (SOTA) methods on different datasets. We typically train our full model on FSOD training set and directly evaluate on these datasets. For fair comparison with other methods, we may discard training on FSOD and adopt the same train/test setting as these methods. In these cases, we use a multi-way[1] few-shot training in the fine-tuning stage with more details to be described.

Training Strategy	Attention RPN	AP_{50}	AP_{75}
1-way 1-shot		55.0	39.1
1-way 1-shot	✓	55.9	41.1
2-way 1-shot		63.8	42.9
2-way 5-shot		65.4	43.7
2-way 5-shot	✓	**67.5**	**46.2**
5-way 5-shot	✓	66.9	45.6

Table 3. Experimental results for training strategy and attention RPN with the multi-relation detector.

5.1. Training Details

Our model is trained end-to-end on 4 Tesla P40 GPUs using SGD with a batch size of 4 (for query images). The learning rate is 0.002 for the first 56000 iterations and 0.0002 for later 4000 iterations. We observe that pre-training on ImageNet [56] and MS COCO [13] can provide stable low-level features and lead to a better converge point. Given this, we by default train our model from the pre-trained ResNet50 on [13, 56] unless otherwise stated. During training, we find that more training iterations may damage performance, where too many training iterations make the model over-fit to the training set. We fix the weights of Res1-3 blocks and only train high-level layers to utilize low-level basic features and avoid over-fitting. The shorter side of the query image is resized to 600 pixels; the longer side is capped at 1000. The support image is cropped around the target object with 16-pixel image context, zero-padded and then resized to a square image of 320 × 320. For few-shot training and testing, we fuse feature by averaging the object features with the same category and then feed them to the attention RPN and the multi-relation detector. We adopt the typical metrics [21], i.e. AP, AP_{50} and AP_{75} for evaluation.

5.2. Comparison with State-of-the-Art Methods

5.2.1 ImageNet Detection dataset

In Table 4, we compare our results with those of LSTD [9] and RepMet [61] on the challenging ImageNet based 50-way 5-shot detection scenario. For fair comparison, we use their evaluation protocol and testing dataset and we use the same MS COCO training set to train our model. We also use soft-NMS [63] as RepMet during evaluation. Our approach produces 1.7% performance gain compared to the state-of-the-art (SOTA) on the AP_{50} evaluation.

To show the generalization ability of our approach, we directly apply our model trained on FSOD dataset on the test set and we obtain 41.7% on the AP_{50} evaluation which is surprisingly better than our fine-tuned model (Table 4). It should be noted that our model trained on FSOD dataset

[1] The fine-tuning stage benefits from more ways during the multi-way training, so we use as many ways as possible to fill up the GPU memory.

Method	dataset	fine-tune	AP_{50}	AP_{75}
LSTD [9]	COCO	$\checkmark^{ImageNet}$	37.4	-
RepMet [11]	COCO	$\checkmark^{ImageNet}$	39.6	-
Ours	COCO	$\checkmark^{ImageNet}$	41.3	21.9
Ours	FSOD†	✗	41.7	28.3
Ours	FSOD†	$\checkmark^{ImageNet}$	**44.1**	**31.0**

Table 4. Experimental results on ImageNet Detection dataset for 50 novel categories with 5 supports. † means that the testing categories are removed from FSOD training dataset. $\checkmark^{ImageNet}$ means the model is fine-tuned on ImageNet Detection dataset.

Method	dataset	fine-tune	AP	AP_{50}	AP_{75}
FR [10]	COCO	$\checkmark^{coco}$	5.6	12.3	4.6
Meta [12]	COCO	$\checkmark^{coco}$	8.7	19.1	6.6
Ours	COCO	$\checkmark^{coco}$	11.1	20.4	10.6
Ours	FSOD†	✗	**16.6**	**31.3**	**16.1**

Table 5. Experimental results on MS COCO minival set for 20 novel categories with 10 supports. † means that the testing categories are removed from FSOD training dataset. $\checkmark^{coco}$ means the model is fine-tuned on MS COCO dataset.

can be directly applied on the test set without fine-tuning to achieve SOTA performance. Furthermore, although our model trained on FSOD dataset has a slightly better AP_{50} performance than our fine-tuned model on the MS COCO dataset, our model surpasses the fine-tuned model by 6.4% on the AP_{75} evaluation, which shows that our proposed FSOD dataset significantly benefits few-shot object detection. With further fine-tuning our FSOD trained model on the test set, our model achieves the best performance, while noting that our method without fine-tuning already works best compared with SOTA.

5.2.2 MS COCO dataset

In Table 5, we compare our approach[1] with Feature Reweighting [10] and Meta R-CNN [12] on MS COCO minival set. We follow their data split and use the same evaluation protocol: we set the 20 categories included in PASCAL VOC as novel categories for evaluation, and use the rest 60 categories in MS COCO as training categories. Our fine-tuned model with the same MS COCO training dataset outperforms Meta R-CNN by 2.4%/1.3%/4.0% on $AP/AP_{50}/AP_{75}$ metrics. This demonstrates the strong learning and generalization ability of our model, as well as that, in the few-shot scenario, learning general matching relationship is more promising than the attempt to learn category-specific embeddings [10, 12]. Our model trained on FSOD achieves more significant improvement of 7.9%/12.2%/9.5% on $AP/AP_{50}/AP_{75}$ metrics. Note that our model trained on FSOD dataset are directly applied on the novel categories without any further fine-tuning while all other methods use 10 supports for fine-tuning to adapt to the novel categories. Again, without fine-tuning our FSOD-trained model already works the best among SOTAs.

Method	FSOD pretrain	fine-tune	AP_{50}	AP_{75}
FRCNN [25]	✗	$\checkmark^{fsod}$	11.8	6.7
FRCNN [25]	✓	$\checkmark^{fsod}$	23.0	12.9
LSTD [9]	✓	$\checkmark^{fsod}$	24.2	13.5
Ours	trained directly	✗	**27.5**	**19.4**

Table 6. Experimental results on FSOD test set for 200 novel categories with 5 supports evaluated in novel category detection. $\checkmark^{fsod}$ means the model is fine-tuned on FSOD dataset.

5.3. Realistic Applications

We apply our approach in different real-world application scenarios to demonstrate its generalization capability. Fig. 8 shows qualitative 1-shot object detection results on novel categories in our test set. We further apply our approach on the wild penguin detection [64] and show sample qualitative 5-shot object detection results in Fig. 9.

Novel Category Detection. Consider this common real-world application scenario: given a massive number of images in a photo album or TV drama series without any labels, the task is to annotate a novel target object (e.g., a rocket) in the given massive collection without knowing which images contain the target object, which can be in different sizes and locations if present. In order to reduce manual labor, one solution is to manually find a small number of images containing the target object, annotate them, and then apply our method to automatically annotate the rest in the image collection. Following this setting, we perform the evaluation as follows: We mix all test images of FSOD dataset, and for each object category, we pick 5 images that contain the target object to perform this novel category object detection in the entire test set. Note that different from the standard object detection evaluation, in this evaluation, the model evaluates every category separately and has no knowledge of the complete categories.

We compare with LSTD [9] which needs to be trained on novel categories by transferring knowledge from the source to target domain. Our method, however, can be applied to detect object in novel categories **without any further re-training or fine-tuning**, which is fundamentally different from LSTD. To compare empirically, we adjust LSTD to base on Faster R-CNN and re-train it on 5 fixed supports for each test category separately in a fair configuration. Results are shown in Table 6. Our method outperforms LSTD by 3.3%/5.9% and its backbone Faster R-CNN by 4.5%/6.5% on all 200 testing categories on AP_{50}/AP_{75} metrics. More specifically, without pre-training on our dataset, the performance of Faster R-CNN significantly drops. Note that because the model only knows the support category, the fine-tuning based models need to train every category separately which is time-consuming.

Wild Car Detection. We apply our method[2] to wild car detection on KITTI [52] and Cityscapes [65] datasets which are urban scene datasets for driving applications, where the images are captured by car-mounted video cameras. We

[1] Since Feature Reweighting and Meta R-CNN are evaluated on MS COCO, in this subsection we discard pre-training on [13] for fair comparison to follow the same experimental setting as described.

[2] We also discard the MS COCO pretraining in this experiment.

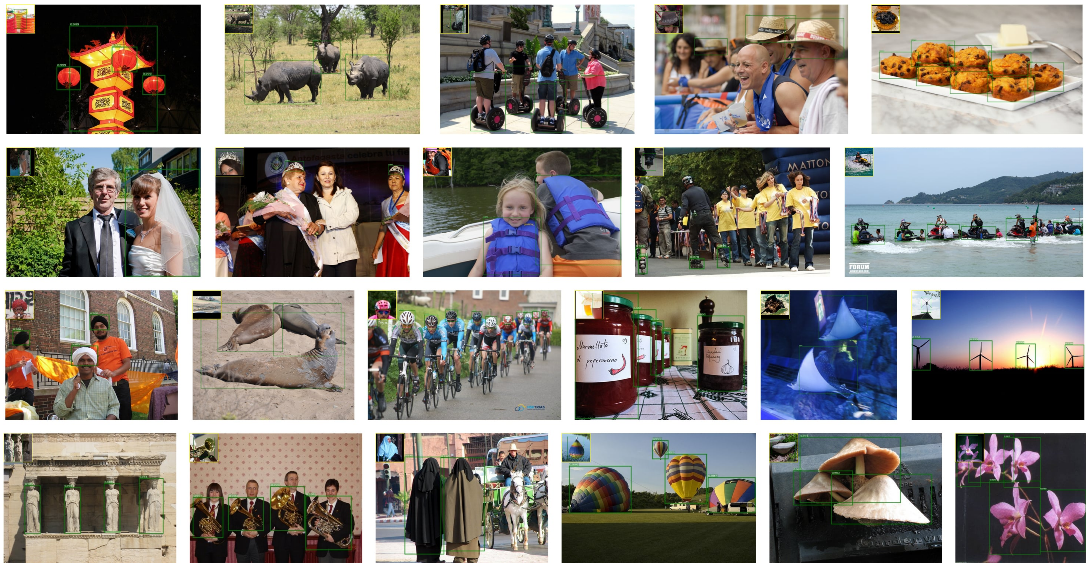

Figure 8. Qualitative 1-shot detection results of our approach on FSOD test set. Zoom in the figures for more visual details.

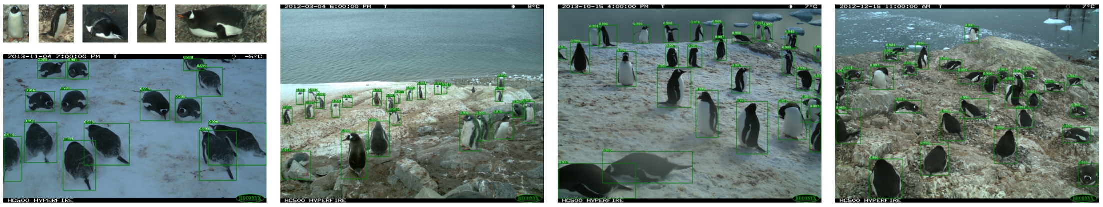

Figure 9. Our application results on the penguin dataset [64]. Given 5 penguin images as support, our approach can detect all penguins in the wild in the given query image.

evaluate the performance of *Car* category on KITTI training set with 7481 images and Cityscapes validation set with 500 images. DA Faster R-CNN [66] uses massively annotated data from source domains (KITTI/Cityscapes) and unlabeled data from target domains (Cityscapes/KITTI) to train the domain adaptive Faster R-CNN, and evaluated the performance on target domains. Without any further retraining or fine-tuning, our model with 10-shot supports obtains comparable or even better AP_{50} performance (37.0% *vs*. 38.5% on Cityscapes and 67.4% *vs*. 64.1% on KITTI) on the wild car detection task. Note that DA Faster R-CNN are specifically designed for the wild car detection task and they use much more training data in similar domains.

5.4. More Categories *vs*. More Samples?

Our proposed dataset has a large number of object categories but with few image samples in each category, which we claim is beneficial to few-shot object detection. To confirm this benefit, we train our model on MS COCO dataset, which has more than 115,000 images with only 80 categories. Then we train our model on FSOD dataset with different category numbers while keeping similar number of training image. Table 7 summarizes the experimental results, where we find that although MS COCO has the most training images but its model performance turns out to be the worst, while models trained on FSOD dataset have better performance as the number of categories incrementally increases while keeping similar number of training images, indicating that a limited number of categories with too many images can actually impede few-shot object detection, while large number of categories can consistently benefit the task. Thus, we conclude that category diversity is essential to few-shot object detection.

Dataset	No. Class	No. Image	AP_{50}	AP_{75}
COCO [13]	80	115k	49.1	28.9
FSOD	300	26k	60.3	39.1
FSOD	500	26k	62.7	41.9
FSOD	800	27k	**64.7**	**42.6**

Table 7. Experimental results of our model on FSOD test set with different numbers of training categories and images in the 5-way 5-shot evaluation.

6. Conclusion

We introduce a novel few-shot object detection network with Attention-RPN, Multi-Relation Detectors and Contrastive Training strategy. We contribute a new FSOD which contains 1000 categories of various objects with high-quality annotations. Our model trained on FSOD can detect objects of novel categories requiring no pre-training or further network adaptation. Our model has been validated by extensive quantitative and qualitative results on different datasets. This paper contributes to few-shot object detection and we believe worthwhile and related future work can be spawn from our large-scale FSOD dataset and detection network with the above technical contributions.

References

[1] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, 2017.

[2] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2017.

[3] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. Meta-learning with memory-augmented neural networks. In *ICML*, 2016.

[4] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *NeurIPS*, 2016.

[5] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 2017.

[6] Qi Cai, Yingwei Pan, Ting Yao, Chenggang Yan, and Tao Mei. Memory matching networks for one-shot image recognition. In *CVPR*, 2018.

[7] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *CVPR*, 2018.

[8] 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 *CVPR*, 2018.

[9] Hao Chen, Yali Wang, Guoyou Wang, and Yu Qiao. Lstd: A low-shot transfer detector for object detection. In *AAAI*, 2018.

[10] Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. Few-shot object detection via feature reweighting. In *ICCV*, 2019.

[11] Leonid Karlinsky, Joseph Shtok, Sivan Harary, Eli Schwartz, Amit Aides, Rogerio Feris, Raja Giryes, and Alex M Bronstein. Repmet: Representative-based metric learning for classification and few-shot object detection. In *CVPR*, 2019.

[12] 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 *ICCV*, 2019.

[13] 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 *ECCV*, 2014.

[14] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[15] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2010.

[16] P VIODA. Rapid object detection using a boosted cascade of simple features. In *CVPR*, 2001.

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*. 2012.

[18] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016.

[19] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *CVPR*, 2017.

[20] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *ECCV*, 2016.

[21] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017.

[22] Songtao Liu, Di Huang, et al. Receptive field block net for accurate and fast object detection. In *ECCV*, 2018.

[23] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014.

[24] Ross Girshick. Fast r-cnn. In *ICCV*, 2015.

[25] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015.

[26] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017.

[27] Bharat Singh, Mahyar Najibi, and Larry S Davis. Sniper: Efficient multi-scale training. In *NeurIPS*, 2018.

[28] Sebastian Thrun. Is learning the n-th thing any easier than learning the first? In *NeurIPS*, 1996.

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

[30] Brenden Lake, Ruslan Salakhutdinov, Jason Gross, and Joshua Tenenbaum. One shot learning of simple visual concepts. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 33, 2011.

[31] Brenden M Lake, Ruslan R Salakhutdinov, and Josh Tenenbaum. One-shot learning by inverting a compositional causal process. In *NeurIPS*, 2013.

[32] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

[33] Alex Wong and Alan L Yuille. One shot learning via compositions of meaningful patches. In *ICCV*, 2015.

[34] Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. Tadam: Task dependent adaptive metric for improved few-shot learning. In *NeurIPS*, 2018.

[35] Eleni Triantafillou, Richard Zemel, and Raquel Urtasun. Few-shot learning through an information retrieval lens. In *NeurIPS*, 2017.

[36] Bharath Hariharan and Ross Girshick. Low-shot visual recognition by shrinking and hallucinating features. In *ICCV*, 2017.

[37] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML Workshop*, 2015.

[38] Tsendsuren Munkhdalai and Hong Yu. Meta networks. In *ICML*, 2017.

[39] Tsendsuren Munkhdalai, Xingdi Yuan, Soroush Mehri, and Adam Trischler. Rapid adaptation with conditionally shifted neurons. In *ICML*, 2018.

[40] Yu-Xiong Wang, Ross Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In *CVPR*, 2018.

[41] Wenbin Li, Lei Wang, Jinglin Xu, Jing Huo, Gao Yang, and Jiebo Luo. Revisiting local descriptor based image-to-class measure for few-shot learning. In *CVPR*, 2019.

[42] Yann Lifchitz, Yannis Avrithis, Sylvaine Picard, and Andrei Bursuc. Dense classification and implanting for few-shot learning. In *CVPR*, 2019.

[43] Sungwoong Kim Chang D. Yoo Jongmin Kim, Taesup Kim. Edge-labeling graph neural network for few-shot learning. In *CVPR*, 2019.

[44] Spyros Gidaris and Nikos Komodakis. Generating classification weights with gnn denoising autoencoders for few-shot learning. In *CVPR*, 2019.

[45] Hongyang Li, David Eigen, Samuel Dodge, Matthew Zeiler, and Xiaogang Wang. Finding task-relevant features for few-shot learning by category traversal. In *CVPR*, 2019.

[46] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 2017.

[47] Nanqing Dong and Eric P Xing. Few-shot semantic segmentation with prototype learning. In *BMVC*, 2018.

[48] Claudio Michaelis, Matthias Bethge, and Alexander S. Ecker. One-shot segmentation in clutter. In *ICML*, 2018.

[49] Tao Hu, Pengwan, Chiliang Zhang, Gang Yu, Yadong Mu, and Cees G. M. Snoek. Attention-based multi-context guiding for few-shot semantic segmentation. In *AAAI*, 2019.

[50] Liang-Yan Gui, Yu-Xiong Wang, Deva Ramanan, and José M. F. Moura. Few-shot human motion prediction via meta-learning. In *ECCV*, 2018.

[51] Xuanyi Dong, Liang Zheng, Fan Ma, Yi Yang, and Deyu Meng. Few-example object detection with model communication. *IEEE transactions on pattern analysis and machine intelligence*, 41(7):1641–1654, 2018.

[52] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012.

[53] 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.

[54] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv:1811.00982*, 2018.

[55] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017.

[56] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[57] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[58] Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr. Fully-convolutional siamese networks for object tracking. In *ECCV*, 2016.

[59] Erika Lu, Weidi Xie, and Andrew Zisserman. Class-agnostic counting. In *ACCV*, 2018.

[60] Bo Li, Wei Wu, Qiang Wang, Fangyi Zhang, Junliang Xing, and Junjie Yan. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In *CVPR*, 2019.

[61] Eli Schwartz, Leonid Karlinsky, Joseph Shtok, Sivan Harary, Mattias Marder, Sharathchandra Pankanti, Rogerio Feris, Abhishek Kumar, Raja Giries, and Alex M Bronstein. Repmet: Representative-based metric learning for classification and one-shot object detection. In *CVPR*, 2019.

[62] 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.

[63] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S. Davis. Soft-nms improving object detection with one line of code. In *ICCV*, 2017.

[64] C. Arteta, V. Lempitsky, and A. Zisserman. Counting in the wild. In *ECCV*, 2016.

[65] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016.

[66] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *CVPR*, 2018.