Meta Faster R-CNN: Towards Accurate Few-Shot Object Detection with Attentive Feature Alignment

Guangxing Han, Shiyuan Huang, Jiawei Ma, Yicheng He, Shih-Fu Chang

Columbia University
{gh2561, sh3813, jiawei.m, yh3330, sc250}@columbia.edu

Abstract

Few-shot object detection (FSOD) aims to detect objects using only a few examples. How to adapt state-of-the-art object detectors to the few-shot domain remains challenging. Object proposal is a key ingredient in modern object detectors. However, the quality of proposals generated for few-shot classes using existing methods is far worse than that of many-shot classes, e.g., missing boxes for few-shot classes due to misclassification or inaccurate spatial locations with respect to true objects. To address the noisy proposal problem, we propose a novel meta-learning based FSOD model by jointly optimizing the few-shot proposal generation and fine-grained few-shot proposal classification. To improve proposal generation for few-shot classes, we propose to learn a lightweight metric-learning based prototype matching network, instead of the conventional simple linear object/nonobject classifier, e.g., used in RPN. Our non-linear classifier with the feature fusion network could improve the discriminative prototype matching and the proposal recall for few-shot classes. To improve the fine-grained few-shot proposal classification, we propose a novel attentive feature alignment method to address the spatial misalignment between the noisy proposals and few-shot classes, thus improving the performance of few-shot object detection. Meanwhile we learn a separate Faster R-CNN detection head for many-shot base classes and show strong performance of maintaining base-classes knowledge. Our model achieves state-of-the-art performance on multiple FSOD benchmarks over most of the shots and metrics.

Introduction

Object detection is one of the most fundamental and challenging tasks in computer vision. Recent years have witnessed great progress in this field using deep learning techniques (Ren et al. 2015; Redmon et al. 2016; Liu et al. 2016; He et al. 2017; Tian et al. 2019). However, deep learning based object detection methods need a sufficient amount of human annotations for model training, which are expensive to collect and unavailable for rare categories. Given scarce training data, these models suffer from the risk of overfitting and poor generalization ability (Kang et al. 2019).

This has motivated research on few-shot object detection (FSOD) (Karlinsky et al. 2019). Given a set of base classes with plenty of examples and another set of novel classes with only few examples, the goal is to transfer the knowledge learned from base classes to novel classes to assist in object detection for novel classes. How to learn few-shot object detectors both efficiently and effectively remains challenging.

Copyright © 2022, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

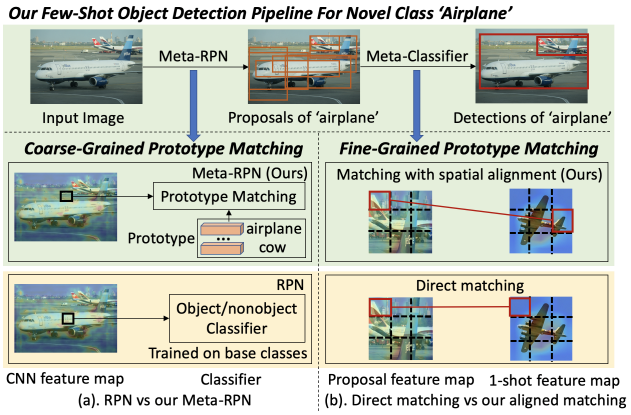

Figure 1: Overview of the proposed model for FSOD. (a) We compare our coarse-grained prototype matching based Meta-RPN with the original RPN. Our method can generate category-specific proposals in an efficient and effective way. (b) Through feature alignment for fine-grained prototype matching, our method can address the spatial misalignment issues between proposals and few-shot classes, leading to better matching and FSOD performance.

Object proposal (Hosang et al. 2015) is a key ingredient in modern object detectors (Ren et al. 2015; Liu et al. 2016), which usually first generate a few potential object bounding boxes (a.k.a proposals) or dense anchor boxes and then convert object detection to a classification task[1]. Training the region proposal network (RPN (Ren et al. 2015)) and proposal classification network (Fast R-CNN (Girshick 2015)) using a large scale dataset have been demonstrated successful. However, as shown in Table 2, the quality of proposals for few-shot novel classes using existing methods are far worse than that of many-shot base classes. Specifically, the detector could miss some high IoU proposals for novel classes due to misclassification with very few examples. Moreover, the

[1]We omit the bounding box regression for simplicity

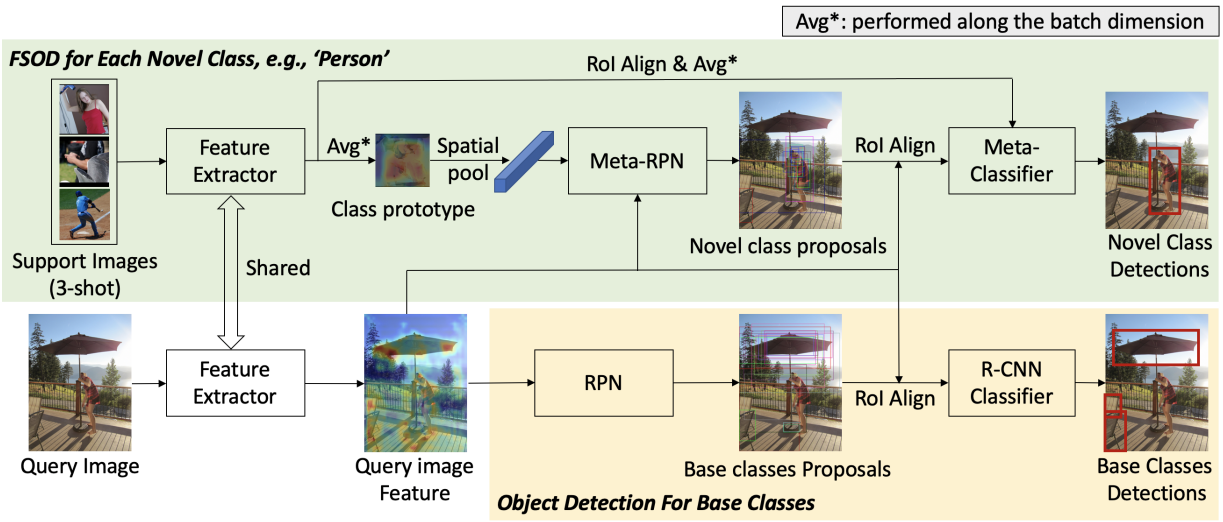

Figure 2: Our proposed object detection model for both many-shot base classes and few-shot novel classes. The detection of base and novel classes are decoupled into two branches with a shared feature extractor. For base classes, we follow the original Faster R-CNN detection pipeline. For novel classes, we propose a novel few-shot detector with a two-stage coarse-to-fine prototype matching network. Our method can improve the accuracy of few-shot object detection for novel classes, and also maintain high accuracy for base classes.

spatial location of novel-class proposals usually do not tightly align with the true objects, and may contain only part of the objects, including large background areas, which makes the following few-shot proposal classification challenging.

To address the aforementioned issues, we propose a meta-learning based FSOD method consisting of two modules, in Figure 1. **First**, the Meta-RPN is proposed to generate category-specific proposals for few-shot novel classes both efficiently and effectively. Specifically, we use a lightweight non-linear matching network to measure the similarity between the dense sliding windows (a.k.a anchor boxes (Ren et al. 2015)) in the query image feature map and the few-shot novel classes, instead of the traditional simple linear object/nonobject classifier, e.g., used in RPN (Ren et al. 2015), thus improving the proposal recall for few-shot novel classes and easing the burden of the following fine-grained proposal classification. **Second**, the Meta-Classifier is proposed to measure the similarity between the noisy proposals and few-shot novel classes. Our key observation is that performing spatial alignment and focusing on corresponding foreground regions between the high-resolution proposal features and class prototypes (Snell, Swersky, and Zemel 2017) is crucial for few-shot proposal classification. To this end, we propose to estimate soft correspondences between each spatial position in the proposal features and class prototypes. Then based on the soft correspondences, we learn to perform spatial alignment between the two features and discover foreground regions. After that, a non-linear prototype matching network is learned to measure the similarity of the aligned features.

The whole network, denoted as Meta Faster R-CNN, can be trained using meta-learning. After meta-training on the data-abundant base classes, our method can enroll few-shot novel classes **incrementally** without any training during meta-testing. Our meta-leaning models achieve competitive results compared with the state-of-the-arts (SOTAs) fine-tuned on novel classes. With further fine-tuning, we can achieve SOTA accuracy on multiple FSOD benchmarks.

To enable base classes detection in our model, previous methods (Wang et al. 2020; Wu et al. 2020; Xiao and Marlet 2020; Yan et al. 2019) usually build a softmax based detector with both base and novel classes. We argue that the softmax detector is inflexible to add new classes because it always needs to fine-tune a new classifier. We also show in Table 6 that our few-shot detector is not suitable for base classes considering both running speed and detection accuracy. Considering both strengths and weaknesses of the two detectors, we propose to take advantage of the two detectors, by learning a separate Faster R-CNN detection head for base classes using the shared feature backbone, as shown in Figure 2. Experimental results demonstrate the effectiveness of our model for both base and novel classes.

Our contributions include: **(1)** We propose to learn a coarse-grained prototype matching network (Meta-RPN) with a metric-learning based non-linear classifier to generate class-specific proposals for novel classes with high recall. **(2)** We propose to learn a fine-grained prototype matching network (Meta-Classifier) with a spatial feature alignment and a foreground attention module to address the spatial misalignment issue between proposal features and class prototypes, which improves the overall detection accuracy. **(3)** Considering both strengths and weaknesses of the softmax based detector and our few-shot detector, we propose to take advantage of the two detectors to detect both base and novel classes by learning two detection heads. **(4)** We achieve SOTA results on multiple FSOD benchmarks.

Related Work

Object Detection. Recently, deep learning (Krizhevsky, Sutskever, and Hinton 2012; He et al. 2016) based methods have dominated the SOTA of object detection. They can be roughly grouped into proposal based methods (Ren et al. 2015; He et al. 2017; Han, Zhang, and Li 2017a, 2018) and proposal-free methods (Redmon et al. 2016; Liu et al. 2016; Han, Zhang, and Li 2017b; Tian et al. 2019). Our method belongs to the first kind as our goal is to push the limit of detection accuracy, which is still the top priority for FSOD.

Few-Shot Object Detection. Building on modern object detectors (e.g., Faster R-CNN (Ren et al. 2015)), existing works (Wang et al. 2020; Wu et al. 2020; Kang et al. 2019; Fan et al. 2020; Perez-Rua et al. 2020; Yan et al. 2019; Xiao and Marlet 2020; Hsieh et al. 2019; Osokin, Sumin, and Lomakin 2020; Wang, Ramanan, and Hebert 2019; Wu, Sahoo, and Hoi 2020; Han et al. 2021, 2022a,b) have explored adapting the current detection pipeline to the few-shot setting, including both proposal generation and proposal classification. For **proposal generation,** many current methods (Karlinsky et al. 2019; Wang et al. 2020) directly use the Region Proposal Network (RPN (Ren et al. 2015)) trained on base classes to generate proposals for novel classes. However, it could miss some high IoU boxes for novel classes as novel-classes boxes are regarded as background regions in RPN training over base classes. Fine-tuning RPN on novel classes (Wu et al. 2020; Xiao and Marlet 2020) could improve the performance, but the generalization ability to unseen classes is limited. Other methods (Kang et al. 2019; Hsieh et al. 2019; Fan et al. 2020) propose to modulate query image features with few-shot classes in order to generate category-specific proposals. However, the simple linear object/nonobject classification in RPNs often lacks the robustness in detecting high-quality proposals needed for FSOD. For **proposal classification and bbox regression,** few-shot learning methods (Vinyals et al. 2016; Snell, Swersky, and Zemel 2017; Sung et al. 2018; Finn, Abbeel, and Levine 2017; Gidaris and Komodakis 2018; Ma et al. 2021; Huang et al. 2022; Ypsilantis et al. 2021), especially prototypical networks (Snell, Swersky, and Zemel 2017) are introduced to extract prototype representation for each class, and then classification can be performed by using a neural network to measure the similarity between proposal features and class prototypes (Sung et al. 2018; Koch, Zemel, and Salakhutdinov 2015). This has been demonstrated effective in FSOD. However, they ignore the spatial misalignment issue: similar semantic regions do not appear at the same spatial position between the noisy proposals and few-shot support images. Our proposed model with Meta-RPN and Meta-Classifier could alleviate the above-mentioned issues.

Our Approach

Task Definition

In few-shot object detection task, we have two disjoint sets of classes $\mathcal{C} = \mathcal{C}_{base} \cup \mathcal{C}_{novel}$, including base classes $\mathcal{C}_{base}$ and novel classes $\mathcal{C}_{novel}$, and $\mathcal{C}_{base} \cap \mathcal{C}_{novel} = \emptyset$.

For the base classes, we have plenty of labeled training images $\mathcal{D} = \{(I, y), I \in \mathcal{I}, y \in \mathcal{Y}\}$ with bounding box annotations, where I is a training image, y is the ground-truth labels for I. Specifically, $y = \{(c_i, box_i), c_i \in \mathcal{C}_{base}, box_i = \{x_i, y_i, w_i, h_i\}\}_{i=1}^{N}$, containing N bounding boxes in the image I with both class label c_i and box location box_i for each bounding box.

For the novel classes, also known as support classes, we only have K-shot (e.g., $K = 1, 5, 10$) labeled samples for each class, also known as support images. Specifically, the support images for novel class $c \in \mathcal{C}_{novel}$ is $\mathcal{S}_c = \{(I_i^c, box_i), I_i^c \in \mathcal{I}, box_i = \{x_i, y_i, w_i, h_i\}\}_{i=1}^{K}$, where I_i^c is a training image, box_i is the box location of the object with class label c.

The goal is to detect objects of novel classes using few-shot examples and also keep high accuracy for base classes.

The Model Architecture

The key idea of the meta-learning based FSOD is to learn how to match the few-shot classes with query images using the abundant training data of base classes, so that it can generalize to few-shot novel classes. Considering the difficulty of object detection with few-shot examples, we propose to learn the detection model via a coarse-to-fine manner following (Ren et al. 2015). Meanwhile, we learn a separate Faster R-CNN detection head for base classes using plenty of training samples. Our model mainly has the following four modules, as shown in Figure 2.

(1) Feature Extraction. We use a siamese network to extract features for both query images and the support images. Formally, given a query image $I_q \in \mathbb{R}^{H_q \times W_q \times 3}$, a deep feature backbone is used to extract CNN features $f_q = F(I_q) \in \mathbb{R}^{H \times W \times C}$, where H, W, C are the height, width and channel dimension of the extracted features. Typically we use the output after $res4$ block in ResNet-50/101 (He et al. 2016) as our default image features.

For the support images, we first extend the original object bounding boxes with some surrounding context regions as the common practice in previous works (Fan et al. 2020; Kang et al. 2019; Yan et al. 2019; Xiao and Marlet 2020), and crop the corresponding object regions in the image. Then the cropped images are adjusted to the same size (Fan et al. 2020), and fed into the shared feature backbone to extract the CNN features $F(I_i^c)$ for each support image I_i^c.

(2) Object Detection for Base Classes. On top of the feature extraction network, RPN is used to generate category-agnostic proposals of all base classes in the image. After that, for each proposal, an R-CNN classifier (Girshick 2015) is employed to produce softmax probabilities over all base classes $\mathcal{C}_{base}$ plus a "background" class and bbox regression.

(3) Proposal Generation for Novel Classes. We aim to generate proposals for few-shot novel classes both efficiently and effectively. In Figure 3, similar to RPN, we attach a small subnetwork on top of f_q to generate proposals centered at each spatial position of f_q. Specifically, a 3×3 conv and ReLU layer is used to extract features for multi-scale anchors centered at each spatial position. For each novel class, We take the averaged CNN features of the K-shot support images as the class prototype $\{f^c = \frac{1}{K}\sum_{i=1}^{K} F(I_i^c), c \in \mathcal{C}_{novel}\}$. Then in order to get the same feature size as the anchor boxes, we conduct spatial average pooling to get the pooled

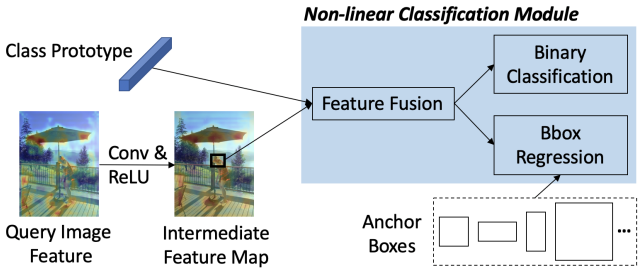

Figure 3: Our proposed Meta-RPN with a non-linear classification module.

prototype $f^c_{pool} = \frac{1}{H*W}\sum_{h,w} f^c$ for each novel class, such that $f^c_{pool} \in \mathbb{R}^C$ captures global representation of class c.

Then intead of using a **simple linear object/nonobject classifier in RPN**, we propose to build a **metric-learning based non-linear classifier** with a lightweight feature fusion network to better calculate the similarity between the class prototypes and anchor features. Based on previous work (Fan et al. 2020; Xiao and Marlet 2020), we propose a stronger feature fusion network with Multiplication (Mult), Subtraction (Sub), and Concatenation (Cat) subnetworks. In the first two subnetworks, we perform the basic element-wise fusion operations. Mult can highlight relevant and common features, and Sub can directly measure the distance between two inputs. Cat can be seen as a learnable operation. Although Sub can be learned by Cat and a convolutional layer, we show by experiments that directly learning a Cat subnetwork is difficult while the other two subnetworks can act as shortcuts, and all the three subnetworks are complementary. Formally,

$$f^c_q = [\Phi_{Mult}(f_q \odot f^c_{pool}), \Phi_{Sub}(f_q - f^c_{pool}), \Phi_{Cat}[f_q, f^c_{pool}]] \quad (1)$$

where Φ_{Mult}, Φ_{Sub} and Φ_{Cat} all consist of a conv and ReLU layer, and $[,]$ denotes channel-wise concatenation. Then f^c_q is fed into a binary classification and bbox regression layer to predict proposals. The proposed feature fusion network can be naturally implemented with convolutional layers, and is computationally efficient and achieve high recall for novel classes using a few proposals.

(4) Proposal Classification and Refinement for Novel Classes. To calculate the similarity of the generated proposals with novel classes, we first use a siamese network (including RoIAlign (He et al. 2017) and the $res5$ block in ResNet) to extract proposal features f_p and class prototype $\hat{f}^c$ of the same size $\mathbb{R}^{H' \times W' \times C'}$, where $H' = W' = 7, C' = 2048$. Different from using the spatially-pooled class prototypes in Meta-RPN, we use high-resolution features for fine-grained matching thanks to the cascade design. However, due to the inaccurate localization of proposals for novel classes, the **spatial misalignment** between proposals and class prototypes have a negative effect on the few-shot classification.

We propose an attention based feature alignment method to address the spatial misalignment issue. As illustrated in Figure 4, we first establish soft correspondences between two input features by calculating the affinity matrix. The affinity matrix is then used to compute proposal-aligned prototype

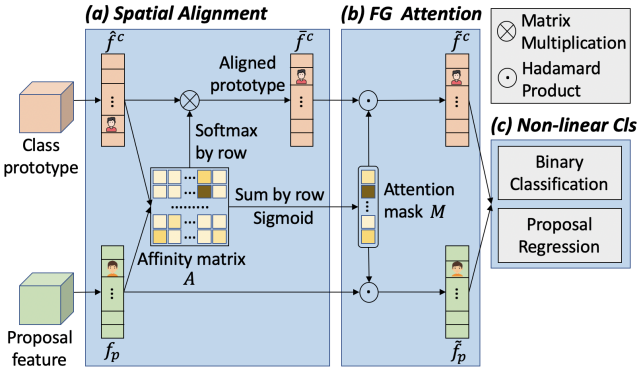

Figure 4: Our proposed Meta-Classifier with Attentive Feature Alignment. To calculate the similarity between the class prototype and proposal feature, we first calculate the proposal-aligned class prototype using the spatial alignment module. Then an attention module is proposed to discover foreground regions. Finally, a non-linear classification module is followed to calculate the similarity.

and localize foreground regions. A non-linear classification module is then followed to calculate the similarity score.

(a) Spatial Alignment Module. Formally, given a pair of input features f_p and $\hat{f}^c$, both feature maps have $H' \times W'$ CNN grid features, each with a embedding dimension of $\mathbb{R}^{C'}$. We compute the affinity matrix $A \in \mathbb{R}^{H'W' \times H'W'}$, with each item the dot product of two embeddings,

$$A(i,j) = f_p(i)\hat{f}^c(j)^\intercal \quad (2)$$

where $A(i,j)$ denotes the similarity of embeddings $f_p(i)$ and $\hat{f}^c(j)$ at spatial location i and j of the proposal feature and class prototype respectively. Next is to calculate the proposal-aligned class prototype. The feature alignment is achieved by taking the weighted average of all CNN grid features in the prototype according to the dense semantic correspondence in the affinity matrix A. An example is shown in Figure 4. We show the importance of the alignment direction in Table 4.

Formally, for a spatial location $i \in \mathbb{R}^{H' \times W'}$ in f_p, softmax normalization is performed over all spatial locations in $\hat{f}^c(j)$,

$$A(i,j)' = \frac{\exp(A(i,j))}{\sum_k \exp(A(i,k))} \quad (3)$$

Then, the proposal-aligned class prototype $\bar{f}^c$ at each spatial location $i \in \mathbb{R}^{H' \times W'}$, can be calculated by aggregating the embeddings of all positions in class prototype $\hat{f}^c$ using the normalized similarity,

$$\bar{f}^c(i) = \sum_j A(i,j)' \hat{f}^c(j) \quad (4)$$

(b) Foreground Attention Module. As the proposals may contain undesired background regions, a foreground attention mask $M \in \mathbb{R}^{H' \times W'}$ is generated to highlight the corresponding object regions. Formally, for each spatial location i in f_p,

we summarize the similarity of $f_p(i)$ to each spatial location in $\hat{f}^c$ using the affinity matrix A. Then sigmoid function (σ) is applied to get the normalized probability,

$$M(i) = \sigma(\sum_j A(i,j)) = \frac{1}{1+\exp(-\sum_j A(i,j))} \quad (5)$$

where higher values in M indicate that the corresponding locations in f_p are more similar to that of the aligned prototype $\bar{f}^c$, and are more likely to be the same semantic parts. On the other hand, the background regions in the proposals can hardly find corresponding locations in the class prototypes with high similarity, leading to lower values in M. Therefore we multiply the attention mask M with both f_p and $\bar{f}^c$ to focus on the corresponding foreground regions,

$$\grave{f}^c = M \odot \bar{f}^c, \quad \grave{f}_p = M \odot f_p \quad (6)$$

In addition, we further multiply the features $\grave{f}^c$ and $\grave{f}_p$ with learnable parameters γ_1 and γ_2 (both initialized as 0) and add back the input features for stable training,

$$\widetilde{f}^c = \gamma_1 \grave{f}^c + \hat{f}^c, \quad \widetilde{f}_p = \gamma_2 \grave{f}_p + f_p \quad (7)$$

(c) Non-linear Classification Module. To measure the similarity between the final features $\widetilde{f}^c$ and $\widetilde{f}_p$, we employ a feature fusion network to aggregate the two features with high-resolution,

$$f = [\Psi_{Mult}(\widetilde{f}^c \odot \widetilde{f}_p), \Psi_{Sub}(\widetilde{f}^c - \widetilde{f}_p), \Psi_{Cat}[\widetilde{f}^c, \widetilde{f}_p]] \quad (8)$$

where Ψ_{Mult}, Ψ_{Sub} and Ψ_{Cat} are three similar CNN networks, each with 3 conv and ReLU layers for non-linear fusion. After that, a binary classification and bbox regression layer are followed for final detection.

The Training Framework

Our training framework has the following three steps,

Meta-learning with base classes. We learn our Meta-RPN and Meta-Classifier using meta-training. In each training episode, we sample a few classes from base classes, each class with k-shot support images, to simulate the FSOD scenario for novel classes. Besides, we also sample a few query images with ground-truth boxes, and use binary cross-entropy loss and smooth $L1$ loss (Girshick 2015) for model training. To prevent the vast number of negative matching pairs from overwhelming the training loss, we keep a ratio of 1:3 for positive and negative matching pairs for balanced model training. After meta-training, the model can be directly applied to novel classes during meta-testing without any training.

Learning the separate detection head for base classes. After meta-training, we fix the parameters of the backbone feature extractor, and learn the RPN and R-CNN module for base classes following (Ren et al. 2015).

Fine-tuning with both base and novel classes. We only use base-classes dataset for training in the first two step. For fine-tuning, we sample a small balanced dataset (original images) of both base and novel classes following the common practice in previous works (Kang et al. 2019; Yan et al. 2019; Xiao and Marlet 2020; Fan et al. 2020; Wu et al. 2020; Wang et al. 2020). We make sure that the total number instances for each novel class is exactly k-shot in the sampled dataset, which are also used as the support set during testing.

The key difference of meta-learning and fine-tuning is that, there is no training on novel classes in meta-learning. We only use the support set of novel classes to calculate prototypes during meta-testing. The support images are cropped from the original images using ground-truth annotations. While during fine-tuning, we use the original novel-classes images as query images to fine-tune our few-shot detector, including both the Meta-RPN and Meta-Classifier. The model performance for novel classes will improve when we gradually use more images for fine-tuning.

Experimental Results

Datasets

We use two widely-used FSOD benchmarks MSCOCO (Lin et al. 2014) and PASCAL VOC (Everingham et al. 2010) for model evaluation, and follow FSOD settings the same as previous works (Kang et al. 2019; Wang et al. 2020) by using the exact same few-shot images for fair comparison.

More implementation details are included in the supplementary material.

Ablation Study

Effectiveness of our Meta-RPN. We compare three different proposal generation methods for novel classes (RPN, Attention-RPN (Fan et al. 2020), and our Meta-RPN) in Table 1 (b), (c) and (d) and Table 2. **(1)** The object/nonobject classifier in RPN, pretrained on base classes, do not generalize well to novel classes as only base-classes regions are regarded as true objects in training and it may miss some high IoU boxes for novel classes due to misclassification. Both proposal AR and detection AP of novel classes using RPN are much lower than the others. **(2)** Using class prototype to reweight query image features in Attention-RPN (Fan et al. 2020) can generate class-specific proposals and improve both proposal AR and detection AP. **(3)** By using a metric-learning based non-linear classifier with a lightweight feature fusion network, our Meta-RPN can perform better prototype matching compared with the simple linear object/nonobject classifier used in Attention-RPN (Fan et al. 2020). Our Meta-RPN consistently improves using different number of proposals, especially after fine-tuning. We use 100 proposals for each novel class by default.

Effectiveness of our Meta-Classifier. (1) We first show that using high-resolution features in our Meta-Classifier can achieve good results by comparing Table 1 (a) and (b). This is because fine-grained details could provide important clues when performing matching between two features. **(2)** Although using high-resolution feature maps improve matching performance, the large spatial misalignment between proposals and few-shot classes is harmful for matching due to the inaccurate proposal localization. Using our attentive feature alignment with both spatial alignment and foreground attention module, the performance can be improved consistently for most of the shots and metrics by comparing models in Table 1 (d), (e) and (f). **(3)** We show in Table 4 the importance

	Proposal Generation			Proposal Classification			2-shot			10-shot			30-shot		
	RPN	Attention-RPN†	Meta-RPN	NL-Cls‡	Alignment	FG-Attention	AP	AP50	AP75	AP	AP50	AP75	AP	AP50	AP75
Meta-training the model on base classes, and meta-testing on novel classes															
(a)	✓			✓(LR§)			4.9	10.5	4.1	6.8	13.5	6.0	7.6	15.4	6.6
(b)	✓			✓			5.1	11.0	4.3	7.2	14.5	6.4	8.3	16.9	7.4
(c)		✓		✓			5.5	11.6	4.7	7.7	15.7	6.9	9.0	18.5	7.8
(d)			✓	✓			6.0	12.7	5.0	8.3	16.7	7.3	9.5	19.1	8.4
(e)			✓	✓	✓		6.4	13.6	5.4	8.6	17.5	7.6	9.8	20.0	8.7
(f)			✓	✓	✓	✓	7.0	14.7	5.9	9.2	18.3	8.0	10.2	20.5	9.3
(g)¶			✓	✓	✓	✓	7.0	13.5	**6.4**	9.7	18.5	9.0	11.3	21.2	10.6
Fine-tuning the model on novel classes, and testing on novel classes															
(h)			✓	✓	✓	✓	7.0	15.3	5.7	11.1	23.4	9.5	15.4	31.7	13.4
(i)¶			✓	✓	✓	✓	**7.6**	**16.3**	6.2	**12.7**	**25.7**	**10.8**	**16.6**	**31.8**	**15.8**

Table 1: Ablation study on each component in our approach. We show results of the novel classes on the MSCOCO dataset under shots 2, 10 and 30. † (Fan et al. 2020). ‡ Our Non-Linear Classifier. § Using Low-Resolution features. ¶ Using ResNet-101, otherwise ResNet-50.

#Proposals†	RPN	Attention-RPN‡	Our Meta-RPN
Meta-training the model on base classes, and meta-testing on novel classes			
10	7.8	18.0	**18.3**
100	20.5	32.7	**33.2**
1000	35.0	44.1	**44.9**
Fine-tuning the model on novel classes, and testing on novel classes			
10	9.2	18.7	**18.9**
100	22.9	33.2	**33.8**
1000	37.4	44.7	**45.8**
Base classes: AR@10=27.6, AR@100=43.3, AR@1000=51.0			

Table 2: Proposal average recall (AR) using the models in Table 1 (b), (c) and (d), with 10-shot meta-testing and 10-shot fine-tuning. † The number of proposals generated for each novel class. ‡ (Fan et al. 2020).

Mult	Sub	Cat	Meta-RPN			Meta-Classifier		
			AP	AP50	AP75	AP	AP50	AP75
✓			7.7	16.0	6.9	7.3	15.9	6.1
	✓		7.6	15.8	6.7	4.9	10.2	4.5
		✓	7.4	15.3	6.6	5.3	10.4	5.2
✓	✓		8.0	16.3	7.1	7.6	16.3	6.7
✓		✓	8.1	16.5	7.2	7.9	16.5	7.1
	✓	✓	7.8	16.1	6.9	5.7	10.9	5.4
✓	✓	✓	**8.3**	**16.7**	**7.3**	**8.3**	**16.7**	**7.3**

Table 3: Ablation study on feature fusion, using model in Table 1(d) with only meta-training and 10-shot meta-testing.

Direction	2-shot			10-shot		
	AP	AP50	AP75	AP	AP50	AP75
$\hat{f}^c \leftarrow f_p$	5.1	11.2	4.0	6.5	13.5	5.7
$\hat{f}^c \rightarrow f_p$	**7.0**	**14.7**	**5.9**	**9.2**	**18.3**	**8.0**

Table 4: Ablation study on the direction of feature alignment, using model in Table 1(f) with only meta-training. $\hat{f}^c$ is the novel class prototype, and f_p is the proposal features.

of the alignment direction. We can keep the original structural information in the proposal for precise bbox regression, when aligning the class prototype to the proposal feature. **(4)** We also show that using a stronger feature backbone ResNet-101 can have better results than ResNet-50 in Table 1 (f-g) and (h-i), especially after fine-tuning. **(5)** Visualization of the affinity matrix and attention masks are shown in Figure 5.

Effectiveness of our non-linear feature fusion network. As shown in Table 3, we perform ablation study of our proposed feature fusion network in both Meta-RPN and Meta-Classifier. We can find that using simple element-wise fusion operations especially the Mult subnetwork shows good results, which is widely used in previous work (Kang et al. 2019; Yan et al. 2019; Xiao and Marlet 2020; Fan et al. 2020). Directly using the Cat subnetwork does not achieve good results in both Meta-RPN and Meta-Classifier. This is because the Cat subnetwork attempts to learn complex fusion between the two features, which is not easy for training and generalization. When combining all three subnetworks together, it could ease the training of the Cat subnetwork and learn complementary fusion beyond the Mul and Sub subnetworks.

Effectiveness of both meta-learning and fine-tuning. We show the comparison of meta-training and fine-tuning in Table 1 (f) and (h) for ResNet-50, and in Table 1 (g) and (i) for ResNet-101. We can find that using fine-tuning, the performance improves for large shot settings (e.g., 10/30 shot). However, for extremely few-shot setting (e.g., 2 shot), the performance could hardly improve because fine-tuning is prone to overfitting using very few examples.

We would like to highlight the advantage of our meta-learning method. **(1) Efficient for adaptation to novel classes**. After meta-training on base classes, there is no further training during meta-testing. To enroll novel classes, we only need to calculate their prototype representations via net-

Method	Venue	Novel Set 1					Novel Set 2					Novel Set 3				
		1	2	3	5	10	1	2	3	5	10	1	2	3	5	10
Meta-training the model on base classes, and meta-testing on novel classes																
Fan et al. (Fan et al. 2020)†	CVPR 2020	32.4	22.1	23.1	31.7	35.7	14.8	18.1	24.4	18.6	19.5	25.8	20.9	23.9	27.8	29.0
Meta Faster R-CNN (Ours)	This work	**40.2**	**30.5**	**33.3**	**42.3**	**46.9**	**26.8**	**32.0**	**39.0**	**37.7**	**37.4**	**34.0**	**32.5**	**34.4**	**42.7**	**44.3**
Fine-tuning the model on novel classes, and testing on novel classes																
FSRW (Kang et al. 2019)	ICCV 2019	14.8	15.5	26.7	33.9	47.2	15.7	15.3	22.7	30.1	40.5	21.3	25.6	28.4	42.8	45.9
MetaDet (Wang, Ramanan, and Hebert 2019)	ICCV 2019	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
Meta R-CNN (Yan et al. 2019)	ICCV 2019	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
TFA w/ fc (Wang et al. 2020)	ICML 2020	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 (Wang et al. 2020)	ICML 2020	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
Xiao et al. (Xiao and Marlet 2020)	ECCV 2020	24.2	35.3	42.2	49.1	57.4	21.6	24.6	31.9	37.0	45.7	21.2	30.0	37.2	43.8	49.6
MPSR (Wu et al. 2020)	ECCV 2020	41.7	42.5	51.4	55.2	61.8	24.4	29.3	39.2	39.9	47.8	35.6	41.8	42.3	48.0	49.7
Fan et al. (Fan et al. 2020)†	CVPR 2020	37.8	43.6	51.6	56.5	58.6	22.5	30.6	40.7	43.1	47.6	31.0	37.9	43.7	51.3	49.8
SRR-FSD (Zhu et al. 2021)	CVPR 2021	**47.8**	50.5	51.3	55.2	56.8	**32.5**	35.3	39.1	40.8	43.8	40.1	41.5	44.3	46.9	46.4
TFA + Halluc (Zhang and Wang 2021)	CVPR 2021	45.1	44.0	44.7	55.0	55.9	23.2	27.5	35.1	34.9	39.0	30.5	35.1	41.4	49.0	49.3
CoRPNs + Halluc (Zhang and Wang 2021)	CVPR 2021	47.0	44.9	46.5	54.7	54.7	26.3	31.8	37.4	37.4	41.2	40.4	42.1	43.3	51.4	49.6
FSCE (Sun et al. 2021)	CVPR 2021	44.2	43.8	51.4	61.9	63.4	27.3	29.5	43.5	44.2	50.2	37.2	41.9	47.5	54.6	58.5
Meta Faster R-CNN (Ours)	This work	43.0	**54.5**	**60.6**	**66.1**	**65.4**	27.7	**35.5**	**46.1**	**47.8**	**51.4**	**40.6**	**46.4**	**53.4**	**59.9**	**58.6**

Table 5: Few-shot object detection performance (AP50) on the PASCAL VOC dataset. We use ResNet-101 following most of the previous works. † Our reimplementation results.

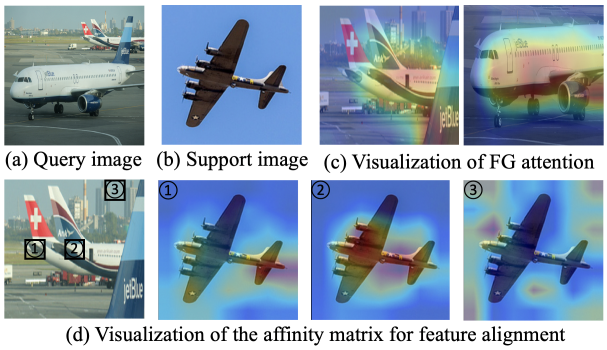

Figure 5: Visualization of the affinity matrix for alignment and foreground attention masks. In (d), we use 3 black boxes in the proposal as query, and show on the right the corresponding similarity maps in the 1-shot support image.

Model	Accuracy			Speed
	AP	AP50	AP75	(/img)
Base model	**39.2**	**59.3**	**42.8**	0.21s
TFA (Wang et al. 2020)	35.8	55.5	39.4	0.21s
2-shot meta-testing	18.5	29.4	19.8	2.1s
10-shot meta-testing	21.4	33.6	23.1	2.1s
30-shot meta-testing	22.7	35.3	24.7	2.1s
Our model	36.9	57.0	40.2	0.21s

Table 6: Base classes evaluation results on MSCOCO using ResNet-101. We compare our final model with the pre-trained base model using the softmax classifier, TFA (Wang et al. 2020) 30-shot fine-tuning, and 2/10/30-shot meta-testing using the few-shot model in Table 1 (g).

work inference. Fine-tuning needs additional training over the original novel-classes images. **(2) Effective for extremely few-shot scenarios**. Our meta-learning model shows superior performance compared with fine-tuning using very few examples (e.g., 1/2 shot in Table 7).

The above two strengths make our meta-learning method promising for realistic applications. We can first perform meta-training using as many base classes as possible. Then the model parameters are fixed. Users can incrementally add new classes to the system using 1-shot support image without forgetting old classes, and without training/fine-tuning.

Effectiveness of learning a separate detection head for base classes. The results are shown in Table 6. **(1)** The pre-trained object detection model on base classes provides an approximate upper bound detection accuracy for base classes. **(2)** We use the learned few-shot object detector for base-classes object detection. Specifically, we select K-shot support images for each base class, and perform object detection for base classes via meta-testing. We find that the meta-testing results on base classes are much higher than novel classes, because the model is meta-trained over base classes, but there is still a big gap compared with the upper bound. The reason is that training a linear classifier with softmax in R-CNN, is more efficient to learn class prototypes than simply averaging the K-shot support images, when the training data is sufficient. Meanwhile, the running speed is slow due to independent detection of each class. **(3)** Our method of learning a separate detection head for base classes, shows better results compared with the strong TFA (Wang et al. 2020) model.

We would like to emphasize the difference between our model and previous methods (Yan et al. 2019; Xiao and Marlet 2020). Although these methods claim to use meta-learning, they still need to train a softmax based detector with both base and novel classes according to their implementation.

Model	Venue	1-shot			2-shot			10-shot			30-shot		
		AP	AP50	AP75	AP	AP50	AP75	AP	AP50	AP75	AP	AP50	AP75
Meta-training the model on base classes, and meta-testing on novel classes													
Fan et al. (Fan et al. 2020)†	CVPR 2020	4.0	8.5	3.5	5.4	11.6	4.6	7.6	15.4	6.8	8.9	17.8	8.0
Meta Faster R-CNN (Ours)	This work	**5.0**	**10.2**	**4.6**	**7.0**	**13.5**	**6.4**	**9.7**	**18.5**	**9.0**	**11.3**	**21.2**	**10.6**
Fine-tuning the model on novel classes, and testing on novel classes													
FSRW (Kang et al. 2019)	ICCV 2019	-	-	-	-	-	-	5.6	12.3	4.6	9.1	19.0	7.6
MetaDet (Wang, Ramanan, and Hebert 2019)	ICCV 2019	-	-	-	-	-	-	7.1	14.6	6.1	11.3	21.7	8.1
Meta R-CNN (Yan et al. 2019)	ICCV 2019	-	-	-	-	-	-	8.7	19.1	6.6	12.4	25.3	10.8
TFA w/ fc (Wang et al. 2020)	ICML 2020	2.9	5.7	2.8	4.3	8.5	4.1	10.0	19.2	9.2	13.4	24.7	13.2
TFA w/ cos (Wang et al. 2020)	ICML 2020	3.4	5.8	3.8	4.6	8.3	4.8	10.0	19.1	9.3	13.7	24.9	13.4
Xiao et al. (Xiao and Marlet 2020)‡	ECCV 2020	3.2	8.9	1.4	4.9	13.3	2.3	10.7	25.6	6.5	15.9	31.7	15.1
MPSR (Wu et al. 2020)	ECCV 2020	2.3	4.1	2.3	3.5	6.3	3.4	9.8	17.9	9.7	14.1	25.4	14.2
Fan et al. (Fan et al. 2020)†	CVPR 2020	4.2	9.1	3.0	5.6	14.0	3.9	9.6	20.7	7.7	13.5	28.5	11.7
SRR-FSD (Zhu et al. 2021)	CVPR 2021	-	-	-	-	-	-	11.3	23.0	9.8	14.7	29.2	13.5
TFA + Halluc (Zhang and Wang 2021)	CVPR 2021	4.4	7.5	**4.9**	5.6	9.9	5.9	-	-	-	-	-	-
CoRPNs + Halluc (Zhang and Wang 2021)	CVPR 2021	3.8	6.5	4.3	5.0	9.0	5.2	-	-	-	-	-	-
FSCE (Sun et al. 2021)	CVPR 2021	-	-	-	-	-	-	11.9	-	10.5	16.4	-	**16.2**
Meta Faster R-CNN (Ours)	This work	**5.1**	**10.7**	4.3	**7.6**	**16.3**	**6.2**	**12.7**	**25.7**	**10.8**	**16.6**	**31.8**	15.8

Table 7: Few-shot object detection performance on the MSCOCO dataset. We use ResNet-101 following most of the previous works. †Our reimplementation results using the exact same few-shot training instances as (Kang et al. 2019; Wang et al. 2020). ‡The authors report these results at https://github.com/YoungXIAO13/FewShotDetection.

Figure 6: Failure cases on COCO dataset. Detection score threshold is 0.5. Left: Missing detection of small objects 'Person'. Right: Misclassification of the 'Boat'. Future work could improve multi-scale object detection and introduce image context information for detection.

The limitation of the softmax based detector is the inflexibility to add new classes to the system, because they need to fine-tune a new softmax based detector with new classes. Our method addresses this problem by decoupling the detection of base and novel classes into two detection heads, and learning a class-agnostic FSOD for novel classes.

Comparison with State-of-the-arts

We compare our final model with the state-of-the-arts on two FSOD benchmarks in Table 5 and 7 respectively. **(1)** Most of the previous works only report fine-tuning results. We demonstrate that our meta-learning-only model improves significantly compared with the strong baseline model (Fan et al. 2020), and outperforms or at least attains comparable results compared with other SOTAs using fine-tuning, especially in extremely few-shot setting (e.g., 1/2-shot), where fine-tuning is not effective. **(2)** We achieve SOTA performance in most of the shots and metrics of the two FSOD benchmarks. The exception is that SRR-FSD (Zhu et al. 2021) and Halluc (Zhang and Wang 2021) have better performance on voc split1&2 1-shot setting due to the introduction of linguistic semantic knowledge and image hallucination, which is complementary to our model. Moreover, our meta-learning-only model shows better results in the more challenging MSCOCO 1/2-shot settings. **(3)** We show the failure cases in Figure 6. More results can be found in the supplementary material.

Conclusion

We propose a novel meta-learning based few-shot object detection model in this paper. Our model consists of the following two modules to tackle with the low quality of proposals for few-shot classes. First, a lightweight coarse-grained prototype matching network is proposed to generate proposals for few-shot classes in an efficient and effective manner. Then, a fine-grained prototype matching network with attentive feature alignment is proposed to address the spatial misalignment between the noisy proposals and few-shot classes. Experiments on multiple FSOD benchmarks demonstrate the effectiveness of our approach.

Acknowledgments

This research is based upon work supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/Interior Business Center (DOI/IBC) contract number D17PC00345. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes not withstanding any copyright annotation theron. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied of IARPA, DOI/IBC or the U.S. Government.

References

Everingham, M.; Van Gool, L.; Williams, C. K.; Winn, J.; and Zisserman, A. 2010. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2): 303–338.

Fan, Q.; Zhuo, W.; Tang, C.-K.; and Tai, Y.-W. 2020. Few-shot object detection with attention-RPN and multi-relation detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4013–4022.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In *International Conference on Machine Learning*, 1126–1135.

Gidaris, S.; and Komodakis, N. 2018. Dynamic few-shot visual learning without forgetting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4367–4375.

Girshick, R. 2015. Fast R-CNN. In *Proceedings of the IEEE international conference on computer vision*, 1440–1448.

Han, G.; He, Y.; Huang, S.; Ma, J.; and Chang, S.-F. 2021. Query Adaptive Few-Shot Object Detection With Heterogeneous Graph Convolutional Networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 3263–3272.

Han, G.; Ma, J.; Huang, S.; Chen, L.; and Chang, S.-F. 2022a. Few-Shot Object Detection with Fully Cross-Transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Han, G.; Ma, J.; Huang, S.; Chen, L.; Chellappa, R.; and Chang, S.-F. 2022b. Multimodal Few-Shot Object Detection with Meta-Learning Based Cross-Modal Prompting.

Han, G.; Zhang, X.; and Li, C. 2017a. Revisiting Faster R-CNN: A Deeper Look at Region Proposal Network. In *International Conference on Neural Information Processing*, 14–24.

Han, G.; Zhang, X.; and Li, C. 2017b. Single shot object detection with top-down refinement. In *2017 IEEE International Conference on Image Processing (ICIP)*, 3360–3364.

Han, G.; Zhang, X.; and Li, C. 2018. Semi-supervised DFF: Decoupling detection and feature flow for video object detectors. In *Proceedings of the 26th ACM international conference on Multimedia*, 1811–1819.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2961–2969.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hosang, J.; Benenson, R.; Dollár, P.; and Schiele, B. 2015. What makes for effective detection proposals? *IEEE transactions on pattern analysis and machine intelligence*, 38(4): 814–830.

Hsieh, T.-I.; Lo, Y.-C.; Chen, H.-T.; and Liu, T.-L. 2019. One-shot object detection with co-attention and co-excitation. In *Advances in Neural Information Processing Systems*, 2725–2734.

Huang, S.; Ma, J.; Han, G.; and Chang, S.-F. 2022. Task-Adaptive Negative Class Envision for Few-Shot Open-Set Recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Kang, B.; Liu, Z.; Wang, X.; Yu, F.; Feng, J.; and Darrell, T. 2019. Few-shot object detection via feature reweighting. In *Proceedings of the IEEE International Conference on Computer Vision*, 8420–8429.

Karlinsky, L.; Shtok, J.; Harary, S.; Schwartz, E.; Aides, A.; Feris, R.; Giryes, R.; and Bronstein, A. M. 2019. Repmet: Representative-based metric learning for classification and few-shot object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5197–5206.

Koch, G.; Zemel, R.; and Salakhutdinov, R. 2015. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2. Lille.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in neural information processing systems*, 1097–1105.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, 740–755. Springer.

Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; and Berg, A. C. 2016. Ssd: Single shot multibox detector. In *European conference on computer vision*, 21–37. Springer.

Ma, J.; Xie, H.; Han, G.; Chang, S.-F.; Galstyan, A.; and Abd-Almageed, W. 2021. Partner-Assisted Learning for Few-Shot Image Classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 10573–10582.

Osokin, A.; Sumin, D.; and Lomakin, V. 2020. OS2D: One-Stage One-Shot Object Detection by Matching Anchor Features. In *European Conference on Computer Vision*.

Perez-Rua, J.-M.; Zhu, X.; Hospedales, T. M.; and Xiang, T. 2020. Incremental Few-Shot Object Detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13846–13855.

Redmon, J.; Divvala, S.; Girshick, R.; and Farhadi, A. 2016. You only look once: Unified, real-time object detection. In *IEEE conference on computer vision and pattern recognition*, 779–788.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, 91–99.

Snell, J.; Swersky, K.; and Zemel, R. 2017. Prototypical networks for few-shot learning. In *Advances in neural information processing systems*, 4077–4087.

Sun, B.; Li, B.; Cai, S.; Yuan, Y.; and Zhang, C. 2021. FSCE: Few-Shot Object Detection via Contrastive Proposal Encoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7352–7362.

Sung, F.; Yang, Y.; Zhang, L.; Xiang, T.; Torr, P. H.; and Hospedales, T. M. 2018. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1199–1208.

Tian, Z.; Shen, C.; Chen, H.; and He, T. 2019. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE international conference on computer vision*, 9627–9636.

Vinyals, O.; Blundell, C.; Lillicrap, T.; Wierstra, D.; et al. 2016. Matching networks for one shot learning. In *Advances in neural information processing systems*, 3630–3638.

Wang, X.; Huang, T. E.; Darrell, T.; Gonzalez, J. E.; and Yu, F. 2020. Frustratingly Simple Few-Shot Object Detection. In *International Conference on Machine Learning (ICML)*.

Wang, Y.-X.; Ramanan, D.; and Hebert, M. 2019. Meta-learning to detect rare objects. In *Proceedings of the IEEE International Conference on Computer Vision*, 9925–9934.

Wu, J.; Liu, S.; Huang, D.; and Wang, Y. 2020. Multi-Scale Positive Sample Refinement for Few-Shot Object Detection. In *European Conference on Computer Vision*, 456–472. Springer.

Wu, X.; Sahoo, D.; and Hoi, S. 2020. Meta-rcnn: Meta learning for few-shot object detection. In *Proceedings of the 28th ACM International Conference on Multimedia*, 1679–1687.

Xiao, Y.; and Marlet, R. 2020. Few-Shot Object Detection and Viewpoint Estimation for Objects in the Wild. In *European Conference on Computer Vision*.

Yan, X.; Chen, Z.; Xu, A.; Wang, X.; Liang, X.; and Lin, L. 2019. Meta r-cnn: Towards general solver for instance-level low-shot learning. In *Proceedings of the IEEE International Conference on Computer Vision*, 9577–9586.

Ypsilantis, N.-A.; Garcia, N.; Han, G.; Ibrahimi, S.; Van Noord, N.; and Tolias, G. 2021. The Met Dataset: Instance-level Recognition for Artworks. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Zhang, W.; and Wang, Y.-X. 2021. Hallucination Improves Few-Shot Object Detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 13008–13017.

Zhu, C.; Chen, F.; Ahmed, U.; Shen, Z.; and Savvides, M. 2021. Semantic Relation Reasoning for Shot-Stable Few-Shot Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8782–8791.