Few-Shot Object Detection and Viewpoint Estimation for Objects in the Wild

Yang Xiao, Renaud Marlet

Few-Shot Object Detection and Viewpoint Estimation for Objects in the Wild

Yang Xiao[1] and Renaud Marlet[1,2]

[1] LIGM, Ecole des Ponts, Univ Gustave Eiffel, CNRS, Marne-la-Valle, France
[2] valeo.ai, Paris, France

Abstract. Detecting objects and estimating their viewpoint in images are key tasks of 3D scene understanding. Recent approaches have achieved excellent results on very large benchmarks for object detection and viewpoint estimation. However, performances are still lagging behind for novel object categories with few samples. In this paper, we tackle the problems of few-shot object detection and few-shot viewpoint estimation. We propose a meta-learning framework that can be applied to both tasks, possibly including 3D data. Our models improve the results on objects of novel classes by leveraging on rich feature information originating from base classes with many samples. A simple joint feature embedding module is proposed to make the most of this feature sharing. Despite its simplicity, our method outperforms state-of-the-art methods by a large margin on a range of datasets, including PASCAL VOC and MS COCO for few-shot object detection, and Pascal3D+ and ObjectNet3D for few-shot viewpoint estimation. And for the first time, we tackle the combination of both few-shot tasks, on ObjectNet3D, showing promising results. Our code and data are available at http://imagine.enpc.fr/~xiaoy/FSDetView/.

Keywords: Few-shot learning, Meta learning, Object detection, Viewpoint estimation.

1 Introduction

Detecting objects in 2D images and estimate their 3D pose, as shown in Fig. 1, is extremely useful for tasks such as 3D scene understanding, augmented reality and robot manipulation. With the emergence of large databases annotated with object bounding boxes and viewpoints, deep-learning-based methods have achieved very good results on both tasks. However these methods, that rely on rich labeled data, usually fail to generalize to *novel* object categories when only a few annotated samples are available. Transferring the knowledge learned from large base categories with abundant annotated images to novel categories with scarce annotated samples is a *few-shot learning* problem.

To address few-shot detection, some approaches simultaneously tackle few-shot classification and few-shot localization by disentangling the learning of

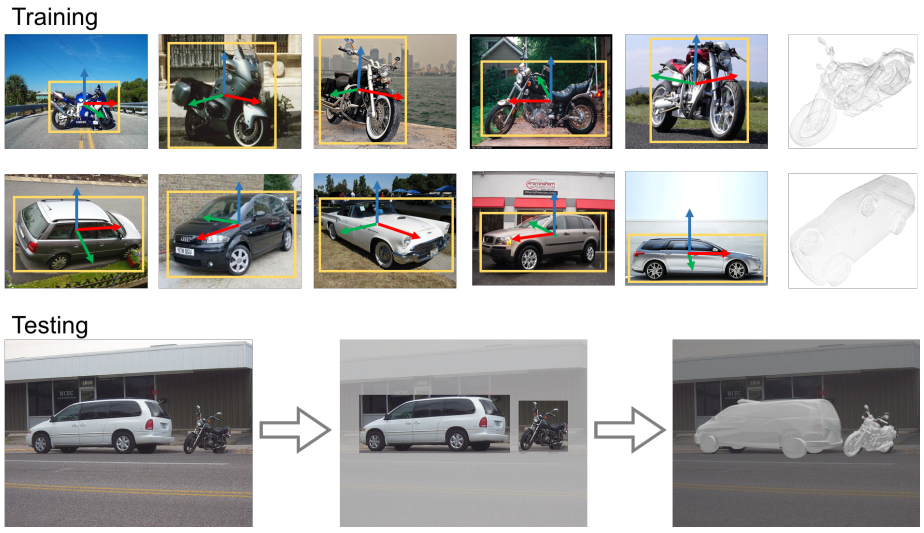


Fig. 1. Few-shot object detection and viewpoint estimation. Starting with images labeled with bounding boxes and viewpoints of objects from base classes, and given only a few similarly labeled images for new categories (top), we predict in a query image the 2D location of objects of new categories, as well as their 3D poses, leveraging on just a few arbitrary 3D class models (bottom).

category-agnostic and category-specific network parameters [59]. Others attach a reweighting module to existing object detection networks [23, 64]. Though these methods have made significant progress, current few-shot detection evaluation protocols suffer from statistical unreliability and the prediction depends heavily on the choice of support data, which makes direct comparison difficult [57].

In parallel to the endeavours made in few-shot object detection, recent work proposes to perform category-agnostic viewpoint estimation that can be directly applied to novel object categories without retraining [65, 63]. However, these methods either require the testing categories to be similar to the training ones [65], or assume the exact CAD model to be provided for each object during inference [63]. Differently, the meta-learning-based method MetaView [53] introduces the category-level few-shot viewpoint estimation problem and addresses it by learning to estimate category-specific keypoints, requiring extra annotations. In any case, precisely annotating the 3D pose of objects in images is far more tedious than annotating their 2D bounding boxes, which makes few-shot viewpoint estimation a non-trivial yet largely under-explored problem.

In this work, we propose a consistent framework to tackle both problems of few-shot object detection and few-shot viewpoint estimation. For this, we exploit, in a meta-learning setting, task-specific class information present in existing datasets, i.e., images with bounding boxes for object detection and, for viewpoint estimation, 3D poses in images as well as a few 3D models for the

different classes. Considering that these few 3D shapes are available is a realistic assumption in most scenarios. Using this information, we obtain an embedding for each class and condition the network prediction on both the class-informative embeddings and instance-wise query image embeddings through a feature aggregation module. Despite its simplicity, this approach leads to a significant performance improvement on novel classes under the few-shot learning regime.

Additionally, by combining our few-shot object detection with our few-shot viewpoint estimation, we address the realistic joint problem of learning to detect objects in images and to estimate their viewpoints from only a few shots. Indeed, compared to other viewpoint estimation methods, that only evaluate in the ideal case with ground-truth (GT) classes and ground-truth bounding boxes, we demonstrate that our few-shot viewpoint estimation method can achieve very good results even based on the predicted classes and bounding boxes.

To summarize, our contributions are:

- We define a simple yet effective unifying framework that tackles both few-shot object detection and few-shot viewpoint estimation.
- We show how to leverage just a few arbitrary 3D models of novel classes to guide and boost few-shot viewpoint estimation.
- Our approach achieves state-of-the-art performance on various benchmarks.
- We propose a few-shot learning evaluation of the new joint task of object detection and view-point estimation, and provide promising results.

2 Related work

Since there is a vast amount of literature on both object detection and viewpoint estimation, we focus here on recent works that target these tasks in the case of limited annotated samples.

Few-shot Learning. Few-shot learning refers to learning from a few labeled training samples per class, which is an important yet unsolved problem in computer vision [28, 16, 56]. One popular solution to this problem is meta-learning [25, 4, 2, 58, 56, 48, 21, 41, 14, 27, 22], where a meta-learner is designed to parameterize the optimization algorithm or predict the network parameters by "learning to learn". Instead of just focusing on the performance improvement on novel classes, some other work has been proposed for providing good results on both base and novel classes [16, 10, 38]. While most existing methods tackle the problem of few-shot image classification, we find that other few-shot learning tasks such as object detection and viewpoint estimation are under-explored.

Object Detection. The general deep-learning models for object detection can be divided into two groups: proposal-based methods and direct methods without proposals. While the R-CNN series [12, 18, 11, 45, 17] and FPN [29] fall into the former line of work, the YOLO series [42, 43, 44] and SSD [31] belong to the latter. All these methods mainly focus on learning from abundant data

to improve detection regarding accuracy and speed. Yet, there are also some attempts to solve the problem with limited labeled data. Chen *et al.* [15] proposes to transfer a pre-trained detector to the few-shot task, while Karlinsky *et al.* [46] exploits distance metric learning to model a multi-modal distribution of each object class.

More recently, Wang *et al.* [59] propose specialized meta-strategies to disentangle the learning of category-agnostic and category-specific components in a detection model. Other approaches based on meta-learning learn a class-attentive vector for each class and use these vectors to reweight full-image features [23] or region-of-interest (RoI) features [64]. Object detection with limited labeled samples is also addressed by approaches targeting weak supervision [49, 5, 6, 47] and zero-shot learning [3, 40, 66], but these settings are different from ours.

Viewpoint Estimation. Deep-learning methods for viewpoint estimation follow roughly three different paths: direct estimation of Euler angles [55, 50, 33, 24, 62, 63], template-based matching [20, 32, 51], and keypoint detection relying on 3D bounding box corners [39, 52, 13, 34, 36] or semantic keypoints [35, 65].

Most of the existing viewpoint estimation methods are designed for known object categories or instances; very little work reports performance on unseen classes [54, 65, 36, 53, 63]. Zhou et al. [65] propose a category-agnostic method to learn general keypoints for both seen and unseen objects, while Xiao et al. [63] show that better results can be obtained when exact 3D models of the objects are additionally provided. In contrast to these category-agnostic methods, Tseng et al. [53] specifically address the few-shot scenario by training a category-specific viewpoint estimation network for novel classes with limited samples.

Instead of using exact 3D object models as [63], we propose a meta-learning approach to extract a class-informative canonical shape feature vector for each novel class from a few labeled samples, with random object models. Besides, our network can be applied to both base and novel classes without changing the network architecture, while [53] requires a separate meta-training procedure for each class and needs keypoint annotations in addition to the viewpoint.

3 Approach

In this section, we first introduce the setup for few-shot object detection and few-shot viewpoint estimation (Sect. 3.1). Then we describe our common network architecture for these two tasks (Sect. 3.2) and the learning procedure (Sect. 3.3).

3.1 Few-shot Learning Setup

We have training samples $(x, y) \in (\mathcal{X}, \mathcal{Y})$ for our two tasks, and a few 3D shapes.

- For object detection, x is an image, $y = \{(\mathsf{cls}_i, \mathsf{box}_i) \mid i \in \mathsf{Obj}_x\}$ indicates the class label cls_i and bounding box box_i of each object i in the image.

Fig. 2. Example of class data for object detection (left) & viewpoint estimation (right).

– For viewpoint estimation, $x = (\mathsf{cls}, \mathsf{box}, \mathsf{img})$ represents an object of class $\mathsf{cls}(x)$ pictured in bounding box $\mathsf{box}(x)$ of an image $\mathsf{img}(x)$, $y = \mathsf{ang} = (\mathsf{azi}, \mathsf{ele}, \mathsf{inp})$ is the 3D pose (viewpoint) of the object, given by Euler angles.

For each class $c \in C = \{\mathsf{cls}_i \mid x \in \mathcal{X}, i \in \mathsf{Obj}_x\}$, we consider a set Z_c of *class data* (see Fig. 2) to learn from using meta-learning:

– For object detection, $Z_c = \{(x, \mathsf{mask}_i) \mid x \in \mathcal{X}, i \in \mathsf{Obj}_x\}$ is made of images x plus an extra channel with a binary mask for bounding box box_i of $i \in \mathsf{Obj}_x$.
– For viewpoint estimation, Z_c is an additional set of 3D models of class c.

At each training iteration, class data z_c is randomly sampled in Z_c for each $c \in C$.

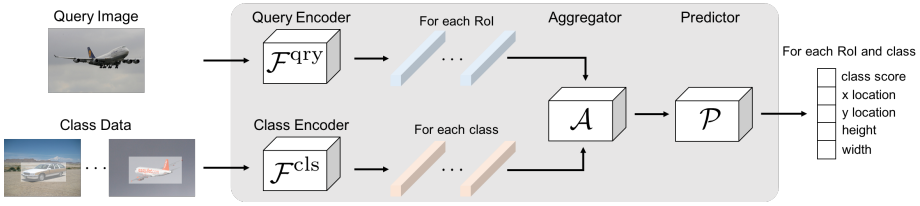
In the few-shot setting, we have a partition of the classes $C = C_{\text{base}} \cup C_{\text{novel}}$ with many samples for base classes in C_{base} and only a few samples (including shapes) for novel classes in C_{novel}. The goal is to transfer the knowledge learned on base classes with abundant samples to little-represented novel classes.

3.2 Network Description

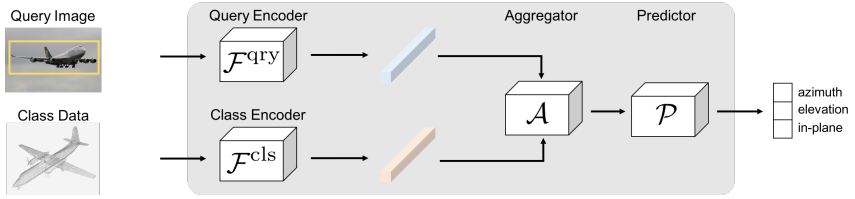
Our general approach has three steps that are visualized in Fig 3. First, query data x and class-informative data z_c pass respectively through the query encoder $\mathcal{F}^{\text{qry}}$ and the class encoder $\mathcal{F}^{\text{cls}}$ to generate corresponding feature vectors. Next, a feature aggregation module $\mathcal{A}$ combines the query features with the class features. Finally, the output of the network is obtained by passing the aggregated features through a task-specific predictor $\mathcal{P}$:

– For object detection, the predictor estimates a classification score and an object location for each region of interest (RoI) and each class.
– For viewpoint estimation, the predictor selects quantized angles by classification, that are refined using regressed angular offsets.

Few-shot object detection. We adopt the widely-used Faster R-CNN [45] approach in our few-shot object detection network (see Fig. 3(a)). The query encoder $\mathcal{F}^{\text{qry}}$ includes the backbone, the region proposal network (RPN) and the proposal-level feature alignment module. In parallel, the class encoder $\mathcal{F}^{\text{cls}}$ is here simply the backbone sharing the same weights as $\mathcal{F}^{\text{qry}}$, that extracts the class features from RGB images sampled in each class, with an extra channel



(a) **Few-shot object detection.**



(b) **Few-shot viewpoint estimation.**

Fig. 3. Method overview.

(a) For object detection, we sample for each class c one image x in the training set containing an object j of class c, to which we add an extra channel for the binary mask mask_j of the ground-truth bounding box box_j of object j. Each corresponding vector of class features $\mathrm{f}_c^{\mathrm{cls}}$ (red) is then combined with each vector of query features $\mathrm{f}_i^{\mathrm{qry}}$ (blue) associated to one of the region of interest i in the query image, via an aggregation module. Finally, the aggregated features $\mathrm{f}_{i,c}^{\mathrm{agg}}$ pass through a predictor that estimates a class probability $\mathsf{cls}_{i,c}$ and regresses a bounding box $\mathsf{box}_{i,c}$.

(b) For few-shot viewpoint estimation, class information is extracted from a few point clouds with coordinates in normalized object canonical space, and the output of the network is the 3D pose represented by three Euler angles.

for a binary mask of the object bounding box [23, 64]. Each extracted vector of query features is aggregated with each extracted vector of class features before being processed for class classification and bounding box regression:

$$\begin{aligned}(\mathsf{cls}_{i,c}, \mathsf{box}_{i,c}) &= \mathcal{P}\Big(\mathcal{A}\big(\mathrm{f}_i^{\mathrm{qry}}, \mathrm{f}_c^{\mathrm{cls}}\big)\Big) \\ \text{for}\quad \mathrm{f}_i^{\mathrm{qry}} &\in \mathcal{F}^{\mathrm{qry}}(x),\ \mathrm{f}_c^{\mathrm{cls}} = \mathcal{F}^{\mathrm{cls}}(z_c),\ c \in C_{\mathrm{train}}\end{aligned} \tag{1}$$

where C_{train} is the set of all training classes, and where $\mathsf{cls}_{i,c}$ and $\mathsf{box}_{i,c}$ are the predicted classification scores and object locations for the i^{th} RoI in query image x and for class c. The prediction branch in Faster R-CNN is class-specific: the network outputs $N_{\mathrm{train}} = |C_{\mathrm{train}}|$ classification scores and N_{train} box regressions for each RoI. The final predictions are obtained by concatenating all the class-wise network outputs.

Few-shot viewpoint estimation. For few-shot viewpoint estimation, we rely on the recently proposed PoseFromShape [63] architecture to implement our network. To create class data z_c, we transform the 3D models in the dataset into

point clouds by uniformly sampling points on the surface, with coordinates in the normalized object canonical space. The query encoder $\mathcal{F}^{\text{qry}}$ and class encoder $\mathcal{F}^{\text{cls}}$ (cf. Fig. 3(b)) correspond respectively to the image encoder ResNet-18 [19] and shape encoder PointNet [37] in PoseFromShape. By aggregating the query features and class features, we estimate the three Euler angles using a three-layer fully-connected (FC) sub-network as the predictor:

$$\begin{aligned}(\mathsf{azi}, \mathsf{ele}, \mathsf{inp}) &= \mathcal{P}\Big(\mathcal{A}\big(\mathrm{f}^{\text{qry}}, \mathrm{f}^{\text{cls}}\big)\Big) \\ \text{with}\quad \mathrm{f}^{\text{qry}} &= \mathcal{F}^{\text{qry}}(\text{crop}(\mathsf{img}(x), \mathsf{box}(x))),\ \mathrm{f}^{\text{cls}} = \mathcal{F}^{\text{cls}}(z_c),\ c = \mathsf{cls}(x)\end{aligned} \tag{2}$$

where $\text{crop}(\mathsf{img}(x), \mathsf{box}(x))$ indicates that the query features are extracted from the image patch after cropping the object. Unlike the object detection making a prediction for each class and aggregating them together to obtain the final outputs, here we only make the viewpoint prediction for the object class $\mathsf{cls}(x)$ by passing the corresponding class data through the network. We also use the mixed classification-and-regression viewpoint estimator of [63]: the output consists of angular bin classification scores and within-bin offsets for three Euler angles: azimuth (azi), elevation (ele), and in-plane rotation (inp).

Feature aggregation. In recent few-shot object detection methods such as MetaYOLO [23] and Meta R-CNN [64], feature are aggregated by reweighting the query features f^{qry} according to the output f^{cls} of the class encoder $\mathcal{F}^{\text{cls}}$:

$$\mathcal{A}(\mathrm{f}^{\text{qry}}, \mathrm{f}^{\text{cls}}) = \mathrm{f}^{\text{qry}} \otimes \mathrm{f}^{\text{cls}} \tag{3}$$

where $\otimes$ represents channel-wise multiplication and f^{qry} has the same number of channels as f^{cls}. By jointly training the query encoder $\mathcal{F}^{\text{qry}}$ and the class encoder $\mathcal{F}^{\text{cls}}$ with this reweighting module, it is possible to learn to generate meaningful reweighting vectors f^{cls}. ($\mathcal{F}^{\text{qry}}$ and $\mathcal{F}^{\text{cls}}$ actually share their weights, except the first layer [64].)

We choose to rely on a slightly more complex aggregation scheme. The fact is that feature subtraction is a different but also effective way to measure similarity between image features [1, 26]. The image embedding f^{qry} itself, without any reweighting, contains relevant information too. Our aggregation thus concatenates the three forms of the query feature:

$$\mathcal{A}(\mathrm{f}^{\text{qry}}, \mathrm{f}^{\text{cls}}) = [\mathrm{f}^{\text{qry}} \otimes \mathrm{f}^{\text{cls}}, \mathrm{f}^{\text{qry}} - \mathrm{f}^{\text{cls}}, \mathrm{f}^{\text{qry}}] \tag{4}$$

where $[\cdot, \cdot, \cdot]$ represents channel-wise concatenation. The last part of the aggregated features in Eq. (4) is independent of the class data. As observed experimentally (Sect. 4.1), this partial disentanglement does not only improve few-shot detection performance, it also reduces the variation introduced by the randomness of support samples.

3.3 Learning Procedure

The learning consists of two phases: *base-class training* on many samples from base classes ($C_{\text{train}} = C_{\text{base}}$), followed by *few-shot fine-tuning* on a balanced

small set of samples from both base and novel classes ($C_{\text{train}} = C_{\text{base}} \cup C_{\text{novel}}$). In both phases, we optimize the network using the same loss function.

Detection loss function. Following Meta R-CNN [64], we optimize our few-shot object detection network using the same loss function:

$$\mathcal{L} = \mathcal{L}_{\text{rpn}} + \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{loc}} + \mathcal{L}_{\text{meta}} \tag{5}$$

where $\mathcal{L}_{\text{rpn}}$ is applied to the output of the RPN to distinguish foreground from background and refine the proposals, $\mathcal{L}_{\text{cls}}$ is a cross-entropy loss for box classifier, $\mathcal{L}_{\text{loc}}$ is a smoothed-L1 loss for box regression, and $\mathcal{L}_{\text{meta}}$ is a cross-entropy loss encouraging class features to be diverse for different classes [64].

Viewpoint loss function. For the task of viewpoint estimation, we discretize each Euler angle with a bin size of 15 degrees and use the same loss function as PoseFromShape [63] to train the network:

$$\mathcal{L} = \sum_{\theta \in \{\mathsf{azi},\mathsf{ele},\mathsf{inp}\}} \mathcal{L}^{\theta}_{\text{cls}} + \mathcal{L}^{\theta}_{\text{reg}} \tag{6}$$

where $\mathcal{L}^{\theta}_{\text{cls}}$ is a cross-entropy loss for angle bin classification of Euler angle θ, and $\mathcal{L}^{\theta}_{\text{reg}}$ is a smoothed-L1 loss for the regression of offsets relatively to bin centers. Here we remove the meta loss $\mathcal{L}_{\text{meta}}$ used in object detection since we want the network to learn useful inter-class similarities for viewpoint estimation, instead of the inter-class differences for box classification in object detection.

Class data construction. For viewpoint estimation, we make use of all the 3D models available for each class (typically less than 10) during both training stages. By contrast, the class data used in object detection requires the label of object class and location, which is limited by the number of annotated samples for novel classes. Therefore, we use large number of class data for base classes in the base training stage (typically $|Z_c| = 200$, as in Meta R-CNN [64]) and limit its size to the number of shots for both base and novel classes in the K-shot fine-tuning stage ($|Z_c| = K$).

For inference, after learning is finished, we construct once and for all class features, instead of randomly sampling class data from the dataset, as done during training. For each class c, we average all corresponding class features used in the few-shot fine-tuning stage:

$$\mathrm{f}^{\text{cls}}_c = \frac{1}{|Z_c|} \sum_{z_c \in Z_c} \mathcal{F}^{\text{cls}}(z_c). \tag{7}$$

This corresponds to the offline computation of all red feature vectors in Fig. 3(a).

Table 1. Few-shot object detection evaluation on PASCAL VOC. We report the mAP with IoU threshold 0.5 (AP50) under 3 different splits for 5 novel classes with a small number of shots. *Results averaged over multiple random runs.

	Novel Set 1					Novel Set 2					Novel Set 3				
Method \ Shots	1	2	3	5	10	1	2	3	5	10	1	2	3	5	10
LSTD [15]	8.2	1.0	12.4	29.1	38.5	11.4	3.8	5.0	15.7	31.0	12.6	8.5	15.0	27.3	36.3
MetaYOLO [23]	14.8	15.5	26.7	33.9	47.2	15.7	15.2	22.7	30.1	40.5	**21.3**	25.6	28.4	42.8	45.9
MetaDet* [59]	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* [64]	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 [57]	22.9	34.5	40.4	46.7	52.0	16.9	26.4	30.5	34.6	39.7	15.7	27.2	34.7	40.8	44.6
TFA* w/cos [57]	**25.3**	**36.4**	42.1	47.9	52.8	18.3	**27.5**	30.9	34.1	39.5	17.9	27.2	34.3	40.8	45.6
Ours*	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**

Table 2. Few-shot object detection evaluation on MS-COCO. We report the mean Averaged Precision and mean Averaged Recall on the 20 novel classes of COCO. *Results averaged over multiple random runs.

		Average Precision						Average Recall					
Shots	Method	0.5:0.95	0.5	0.75	S	M	L	1	10	100	S	M	L
10	LSTD [15]	3.2	8.1	2.1	0.9	2.0	6.5	7.8	10.4	10.4	1.1	5.6	19.6
	MetaYOLO [23]	5.6	12.3	4.6	0.9	3.5	10.5	10.1	14.3	14.4	1.5	8.4	28.2
	MetaDet* [59]	7.1	14.6	6.1	1.0	4.1	12.2	11.9	15.1	15.5	1.7	9.7	30.1
	Meta R-CNN* [64]	8.7	19.1	6.6	2.3	7.7	14.0	12.6	17.8	17.9	**7.8**	15.6	27.2
	TFA* w/fc [57]	9.1	17.3	8.5	–	–	–	–	–	–	–	–	–
	TFA* w/cos [57]	9.1	17.1	8.8	–	–	–	–	–	–	–	–	–
	Ours*	**12.5**	**27.3**	**9.8**	**2.5**	**13.8**	**19.9**	**20.0**	**25.5**	**25.7**	7.5	**27.6**	**38.9**
30	LSTD [15]	6.7	15.8	5.1	0.4	2.9	12.3	10.9	14.3	14.3	0.9	7.1	27.0
	MetaYOLO [23]	9.1	19.0	7.6	0.8	4.9	16.8	13.2	17.7	17.8	1.5	10.4	33.5
	MetaDet* [59]	11.3	21.7	8.1	1.1	6.2	17.3	14.5	18.9	19.2	1.8	11.1	34.4
	Meta R-CNN* [64]	12.4	25.3	10.8	2.8	11.6	19.0	15.0	21.4	21.7	**8.6**	20.0	32.1
	TFA* w/fc [57]	12.0	22.2	11.8	–	–	–	–	–	–	–	–	–
	TFA* w/cos [57]	12.1	22.0	12.0	–	–	–	–	–	–	–	–	–
	Ours*	**14.7**	**30.6**	**12.2**	**3.2**	**15.2**	**23.8**	**22.0**	**28.2**	**28.4**	8.3	**30.3**	**42.1**

4 Experiments

In this section, we evaluate our approach and compare it with state-of-the-art methods on various benchmarks for few-shot object detection and few-shot viewpoint estimation. For a fair comparison, we use the same splits between base and novel classes [23, 53]. For all the experiments, we run 10 trials with random support data and report the average performance.

4.1 Few-Shot Object Detection

We adopt a well-established evaluation protocol for few-shot object detection [23, 59, 64] and report performance on PASCAL VOC [8, 7] and MS-COCO [30].

Experimental setup. PASCAL VOC 2007 and 2012 consist of 16.5k train-val images and 5k test images covering 20 categories. Consistent with the few-shot learning setup in [23, 59, 64], we use VOC 07 and 12 train-val sets for training and VOC 07 test set for testing. 15 classes are considered as base classes, and the remaining 5 classes as novel classes. For a fair comparison, we consider the same 3 splits as in [23, 59, 64, 57], and for each run we only draw K random shots

Table 3. Ablation study on the feature aggregation scheme. Using the same class splits of PASCAL VOC as in Table 1, we measure the performance of few-shot object detection on the novel classes. We report the average and standard deviation of the AP50 metric over 10 runs. f^{qry} is the query features and f^{cls} is the class features.

	Novel Set 1		Novel Set 2		Novel Set 3	
Method \ Shots	3	10	3	10	3	10
$[f^{qry} \otimes f^{cls}]$	35.0 ± 3.6	51.5 ± 5.8	29.6 ± 3.5	45.4 ± 5.5	27.5 ± 5.2	48.1 ± 5.9
$[f^{qry} \otimes f^{cls}, f^{qry}]$	36.6 ± 7.1	49.6 ± 4.3	27.5 ± 5.7	41.6 ± 3.7	28.7 ± 5.9	44.0 ± 2.7
$[f^{qry} \otimes f^{cls}, f^{qry}, f^{cls}]$	37.6 ± 7.2	54.2 ± 4.9	30.0 ± 2.9	41.0 ± 5.3	33.6 ± 5.0	47.5 ± 2.3
$[f^{qry} \otimes f^{cls}, f^{qry} - f^{cls}]$	39.2 ± 4.5	55.5 ± 3.9	31.7 ± 6.2	45.2 ± 3.3	35.6 ± 5.6	48.9 ± 3.3
$[f^{qry} \otimes f^{cls}, f^{qry} - f^{cls}, f^{qry}]$	$\mathbf{42.2 \pm 2.1}$	$\mathbf{57.4 \pm 2.7}$	$\mathbf{31.9 \pm 2.7}$	$\mathbf{45.7 \pm 1.8}$	$\mathbf{37.2 \pm 3.5}$	$\mathbf{49.6 \pm 2.2}$

from each novel class where $K \in \{1, 2, 3, 5, 10\}$. We report the mean Average Precision (mAP) with intersection over union (IoU) threshold at 0.5 (AP50). For MS-COCO, we set the 20 PASCAL VOC categories as novel classes and the remaining 60 categories as base classes. Following [31, 45], we report standard COCO evaluation metrics on this dataset with $K \in \{10, 30\}$.

Training details. We use the same learning scheme as [64], which uses the SGD optimizer with an initial learning rate of 10^{-3} and a batch size of 4. In the first training stage, we train for 20 epochs and divide the learning rate by 10 after each 5 epochs. In the second stage, we train for 5 epochs with learning rate of 10^{-3} and another 4 epochs with a learning rate of 10^{-4}.

Quantitative results. The results are summarized in Table 1 and 2. Our method outperforms state-of-the-art methods in most cases for the 3 different dataset splits of PASCAL VOC, and it achieves the best results on the 20 novel classes of MS-COCO, which validates the efficacy and generality of our approach. Moreover, our improvements on the difficult COCO dataset (around 3 points in mAP) is much larger than the gap among previous methods. This demonstrates that our approach can generalize well to novel classes even in complex scenarios with ambiguities and occluded objects. By comparing results on objects of different sizes contained in COCO, we find that our approach obtains a much better improvement on medium and large objects while it struggles on small objects.

Different feature aggregations. We analyze the impact of different feature aggregation schemes. For this purpose, we evaluate K-shot object detection on PASCAL VOC with $K \in \{3, 10\}$. The results are reported in Table 3. We can see that our feature aggregation scheme $[f^{qry} \otimes f^{cls}, f^{qry} - f^{cls}, f^{qry}]$ yields the best precision. In particular, although the difference $[f^{qry} - f^{cls}]$ could in theory be learned from the individual feature vectors $[f^{qry}, f^{cls}]$, the network performs better when explicitly provided with their subtraction. Moreover, our aggregation scheme significantly reduces the variance introduced by the random sampling of few-shot support data, which is one of the main issues in few-shot learning.

Table 4. Intra-dataset 10-shot viewpoint estimation evaluation. We report Acc30(↑) / MedErr(↓) on the same 20 novel classes of ObjectNet3D for each method, while 80 are used as base classes. All models are trained and evaluated on ObjectNet3D.

Method	bed	bookshelf	calculator	cellphone	computer	door	f_cabinet
StarMap+F [65]	0.32 / 47.2	0.61 / 21.0	0.26 / 50.6	0.56 / 26.8	0.59 / 24.4	- / -	0.76 / 17.1
StarMap+M [65]	0.32 / 42.2	0.76 / 15.7	0.58 / 26.8	0.59 / 22.2	0.69 / 19.2	- / -	0.76 / 15.5
MetaView [53]	0.36 / 37.5	0.76 / 17.2	**0.92** / 12.3	0.58 / 25.1	0.70 / 22.2	- / -	0.66 / 22.9
Ours	**0.64** / **14.7**	**0.89** / **8.3**	0.90 / **8.3**	**0.63** / **12.7**	**0.84** / **10.5**	**0.90** / **0.9**	**0.84** / **10.5**

Method	guitar	iron	knife	microwave	pen	pot	rifle
StarMap+F [65]	0.54 / 27.9	0.00 / 128	0.05 / 120	0.82 / 19.0	- / -	0.51 / 29.9	0.02 / 100
StarMap+M [65]	0.59 / 21.5	0.00 / 136	0.08 / 117	0.82 / 17.3	- / -	0.51 / 28.2	0.01 / 100
MetaView [53]	0.63 / 24.0	0.20 / 76.9	0.05 / **97.9**	0.77 / 17.9	- / -	0.49 / 31.6	0.21 / **80.9**
Ours	**0.72** / **17.1**	**0.37** / **57.7**	**0.26** / 139	**0.94** / **7.3**	**0.45** / **44.0**	**0.74** / **12.3**	**0.29** / 88.4

Method	shoe	slipper	stove	toilet	tub	wheelchair	TOTAL
StarMap+F [65]	- / -	0.08 / 128	0.80 / 16.1	0.38 / 36.8	0.35 / 39.8	0.18 / 80.4	0.41 / 41.0
StarMap+M [65]	- / -	0.15 / 128	0.83 / 15.6	0.39 / 35.5	0.41 / 38.5	0.24 / 71.5	0.46 / 33.9
MetaView [53]	- / -	0.07 / 115	0.74 / 21.7	0.50 / 32.0	0.29 / 46.5	0.27 / 55.8	0.48 / 31.5
Ours	**0.51** / **29.4**	**0.25** / **96.4**	**0.92** / **9.4**	**0.69** / **17.4**	**0.66** / **15.1**	**0.36** / **64.3**	**0.64** / **15.6**

4.2 Few-Shot Viewpoint Estimation

Following the few-shot viewpoint estimation protocol proposed in [53], we evaluate our method under two settings: *intra*-dataset on ObjectNet3D [60] (reported in Tab. 4) and *inter*-dataset between ObjectNet3D and Pascal3D+ [61] (reported in Tab. 5). In both datasets, the number of available 3D models for each class vary from 2 to 16. We use the most common metrics for evaluation: Acc30, which is the percentage of estimations with a rotational error smaller than 30°, and MedErr, which computes the median rotational error measured in degrees. Complying with previous work [65, 53], we only use the non-occluded and non-truncated objects for evaluation and assume in this subsection that the ground truth classes and bounding boxes are provided at test time.

Training details. The model is trained using the Adam optimizer with a batch size of 16. During the base-class training stage, we train for 150 epochs with a learning rate of 10^{-4}. For few-shot fine-tuning, we train for 50 epochs with learning rate of 10^{-4} and another 50 epochs with a learning rate of 10^{-5}.

Compared methods. For few-shot viewpoint estimation, we compare our method to MetaView [53] and to two adaptations of StarMap [65]. More precisely, the authors of MetaView [53] re-implemented StarMap with one stage of ResNet-18 as the backbone, and trained the network with MAML [9] for a fair comparison in the few-shot regime (entries StarMap+M in Tab. 4-5). They also provided StarMap results by just fine-tuning it on the novel classes using the scarce labeled data (entries StarMap+F in Tab. 4-5).

Intra-dataset evaluation. We follow the protocol of [53, 63] and split the 100 categories of ObjectNet3D into 80 base classes and 20 novel classes. As

Table 5. Inter-dataset 10-shot viewpoint estimation evaluation. We report Acc30(↑) / MedErr(↓) on the 12 novel classes of Pascal3D+, while the 88 base classes are in ObjectNet3D. All models are trained on ObjectNet3D and tested on Pascal3D+.

Method	aero	bike	boat	bottle	bus	car	chair
StarMap+F [65]	0.03 / 102	0.05 / 98.8	0.07 / 98.9	0.48 / 31.9	0.46 / 33.0	0.18 / 80.8	0.22 / 74.6
StarMap+M [65]	0.03 / 99.2	0.08 / 88.4	0.11 / 92.2	0.55 / 28.0	0.49 / 31.0	0.21 / 81.4	0.21 / 80.2
MetaView [53]	0.12 / 104	0.08 / 91.3	0.09 / 108	0.71 / 24.0	0.64 / 22.8	0.22 / 73.3	0.20 / 89.1
Ours	**0.24** / **65.0**	**0.34** / **52.4**	**0.27** / **77.3**	**0.88** / **12.6**	**0.78** / **8.2**	**0.49** / **34.0**	**0.33** / **77.4**

Method	table	mbike	sofa	train	tv	TOTAL
StarMap+F [65]	0.46 / 31.4	0.09 / 91.6	0.32 / 44.7	0.36 / 41.7	0.52 / 29.1	0.25 / 64.7
StarMap+M [65]	0.29 / 36.8	0.11 / 83.5	0.44 / 42.9	0.42 / 33.9	0.64 / 25.3	0.28 / 60.5
MetaView [53]	0.39 / 36.0	0.14 / 74.7	0.29 / 46.2	0.61 / 23.8	0.58 / 26.3	0.33 / 51.3
Ours	**0.60** / **21.2**	**0.41** / **45.2**	**0.58** / **21.3**	**0.71** / **12.6**	**0.78** / **19.1**	**0.52** / **28.3**

shown in Table 4, our model outperforms the recently proposed meta-learning-based method MetaView [53] by a very large margin in overall performance: +16 points in Acc30 and half MedErr (from 31.5° down to 15.6°). Besides, keypoint annotations are not available for some object categories such as door, pen and shoe in ObjectNet3D. This limits the generalization of keypoint-based approaches [65, 53] as they require a set of manually labeled keypoints for network training. By contrast, our model can be trained and evaluated on all object classes of ObjectNet3D as we only rely on the shape pose. More importantly, our model can be directly deployed on different classes using the same architecture, while MetaView learns a set of separate category-specific semantic keypoint detectors for each class. This flexibility suggests that our approach is likely to exploit the similarities between different categories (e.g., bicycle and motorbike) and has more potentials for applications to robotics and augmented reality.

Inter-dataset evaluation. To further evaluate our method in a more practical scenario, we use a source dataset for base classes and another target dataset for novel (disjoint) classes. Using the same split as MetaView [53], we use all 12 categories of Pascal3D+ as novel categories and the remaining 88 categories of ObjectNet3D as base categories. Distinct from the previous intra-dataset experiment that focuses more on the cross-category generalization capacity, this inter-dataset setup also reveals the cross-domain generalization ability.

As shown in Tab. 5, our approach again significantly outperforms StarMap and MetaView. Our overall improvement in inter-dataset evaluation is even larger than in intra-dataset evaluation: we gain +19 points in Acc30 and again divide MedErr by about 2 (from 51.3° down to 28.3°). This indicates that our approach, by leveraging viewpoint-relevant 3D information, not only helps the network generalize to novel classes from the same domain, but also addresses the domain shift issues when trained and evaluated on different datasets.

Visual results. We provide in Fig. 4 visualizations of viewpoint estimation for novel objects on ObjectNet3D and Pascal3D+. We show both success (green boxes) and failure cases (red boxes) to help analyze possible error types. We visualize four categories giving the largest median errors: iron, knife, rifle and

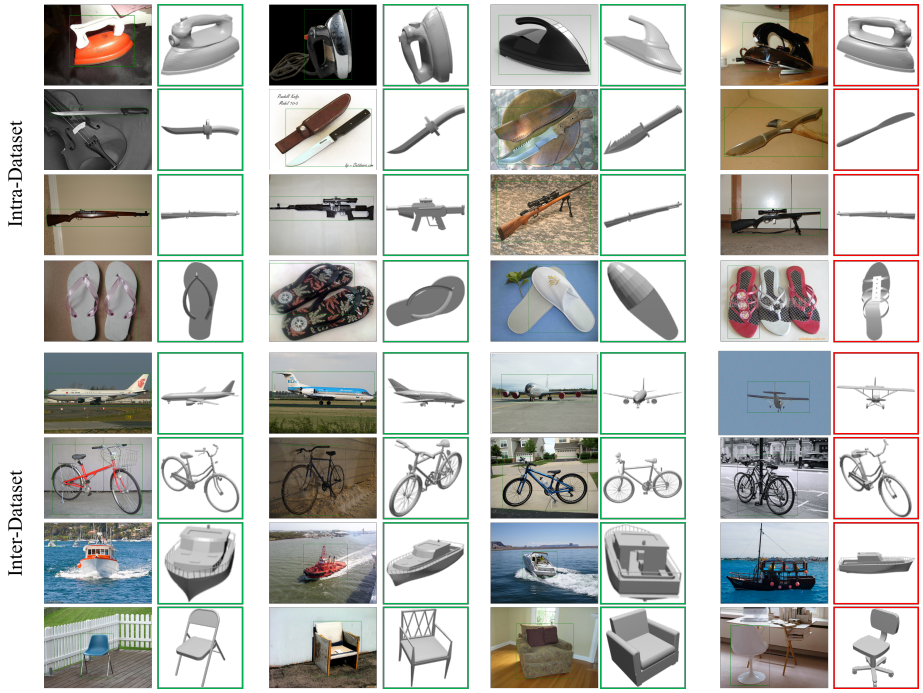


Fig. 4. Qualitative results of few-shot viewpoint estimation. We visualize results on ObjectNet3D and Pascal3D+. For each category, we show three success cases (the first six columns) and one failure case (the last two columns). CAD models are shown here only for the purpose of illustrating the estimated viewpoint.

slipper for ObjectNet3D, and aeroplane, bicycle, boat and chair for Pascal3D+. The most common failure cases come from objects with similar appearances in ambiguous poses, e.g., iron and knife in ObjectNet3D, aeroplane and boat in Pascal3D+. Other failure cases include the heavy clutter cases (bicycle) and large shape variations between training objects and testing objects (chair).

4.3 Evaluation of Joint Detection and Viewpoint Estimation

To further show the generality of our approach in real-world scenarios, we consider the joint problem of detecting objects from novel classes in images and estimating their viewpoints. The fact is that evaluating a viewpoint estimator on ground-truth classes and bounding boxes is a toy setting, not representative of actual needs. On the contrary, estimating viewpoints based on predicted detections is much more realistic and challenging.

To experiment with this scenario, we split ObjectNet3D into 80 base classes and 20 novel classes as in Sect. 4.2, and train the object detector and viewpoint estimator based on the abundant annotated samples for base classes and scarce labeled samples for novel classes. Unfortunately, the codes of StarMap+F/M

Table 6. Evaluation of joint few-shot detection and viewpoint estimation. We report correct prediction percentages on novel classes of ObjectNet3D, first using the ground-truth classes and bounding boxes, then the estimated classes and boxes given by our object detector. Predicted bounding boxes are considered correct with a IoU threshold at 0.5 and estimated viewpoints are considered correct with a rotation error less than 30°. Ours (all-shot) is learned on all training data of the novel classes.

Method	bed	bshelf	calc	cphone	comp	door	fcabin	guit	iron	knife	micro	pen	pot	rifle	shoe	slipper	stove	toilet	tub	wchair	mean
Evaluated using ground-truth classes and bounding boxes (viewpoint estimation)																					
StarMap+M[65]	32	76	58	59	69	–	76	59	0	8	82	–	51	1	–	15	83	39	41	24	46
MetaView[53]	36	76	92	58	70	–	66	63	20	5	77	–	49	21	–	7	74	50	29	27	48
Ours (10-shot)	64	89	90	63	84	90	84	72	37	26	94	45	74	29	51	25	92	69	66	36	64
Ours (all-shot)	81	92	96	65	91	93	89	83	58	28	95	51	81	48	63	53	94	86	77	70	75
Evaluated using predicted classes and bounding boxes (detection + viewpoint estimation)																					
Ours (10-shot)	55	76	74	52	57	69	63	70	44	8	57	22	55	12	6	19	80	65	56	21	48
Ours (all-shot)	65	80	82	56	62	70	66	75	48	9	60	27	61	20	8	32	83	71	67	38	54

and MetaView are not available. The only available information is the results on perfect, ground-truth classes and bounding boxes available in publications. We thus have to reason relatively in terms of baselines. Concretely, we compare these results obtained on ideal input to the case where we use predicted classes and bounding boxes, in the 10-shot scenario. As an upper bound, we also consider the "all-shot" case where all training data of the novel classes are used.

As recalled in Tab. 6, our few-shot viewpoint estimation outperforms other methods by a large margin when evaluated using ground-truth classes and bounding boxes in the 10-shot setting. When using predicted classes and bounding boxes, accuracy drops for most categories. One explanation is that viewpoint estimation becomes difficult when the objects are truncated by imperfect predicted bounding boxes, especially for tiny objects (e.g., shoes) and ambiguous objects with similar appearances in different poses (e.g., knifes, rifles). Yet, by comparing the performance gap between our method when tested using predicted classes and boxes and MetaView when tested using ground-truth classes and boxes, we find that our approach is able to reach the same viewpoint accuracy of 48%, which is a considerable achievement.

5 Conclusion

In this work, we presented an approach to few-shot object detection and viewpoint estimation that can tackle both tasks in a coherent and efficient framework. We demonstrated the benefits of this approach in terms of accuracy, and significantly improved the state of the art on several standard benchmarks for few-shot object detection and few-shot viewpoint estimation. Moreover, we showed that our few-shot viewpoint estimation model can achieve promising results on the novel objects detected by our few-shot detection model, compared to the existing methods tested with ground-truth bounding boxes.

Acknowledgements. We thank Vincent Lepetit and Yuming Du for helpful discussions.

References

1. Ammirato, P., Fu, C.Y., Shvets, M., Kosecka, J., Berg, A.C.: Target driven instance detection (2018), arXiv preprint arXiv:1803.04610
2. Andrychowicz, M., Denil, M., Colmenarejo, S.G., Hoffman, M.W., Pfau, D., Schaul, T., Shillingford, B., de Freitas, N.: Learning to learn by gradient descent by gradient descent. In: International Conference on Neural Information Processing Systems (NeurIPS) (2016)
3. Bansal, A., Sikka, K., Sharma, G., Chellappa, R., Divakaran, A.: Zero-shot object detection. In: European Conference on Computer Vision (ECCV) (2018)
4. Bertinetto, L., Henriques, J.F., Valmadre, J., Torr, P.H.S., Vedaldi, A.: Learning feed-forward one-shot learners. In: International Conference on Neural Information Processing Systems (NeurIPS) (2016)
5. Bilen, H., Vedaldi, A.: Weakly supervised deep detection networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2846–2854 (2016)
6. Diba, A., Sharma, V., Pazandeh, A.M., Pirsiavash, H., Gool, L.V.: Weakly supervised cascaded convolutional networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5131–5139 (2017)
7. Everingham, M., Eslami, S.M.A., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL visual object classes challenge: A retrospective. International Journal of Computer Vision (IJCV) (2015)
8. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL visual object classes (voc) challenge. International Journal of Computer Vision (IJCV) (2010)
9. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: International Conference on Machine Learning (ICML) (2017)
10. Gidaris, S., Komodakis, N.: Dynamic few-shot visual learning without forgetting. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4367–4375 (2018)
11. Girshick, R.: Fast R-CNN. In: IEEE International Conference on Computer Vision (ICCV) (2015)
12. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2014)
13. Grabner, A., Roth, P.M., Lepetit, V.: 3D pose estimation and 3D model retrieval for objects in the wild. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3022–3031 (2018)
14. Ha, D., Dai, A., Le, Q.V.: HyperNetworks. In: International Conference on Learning Representations (ICLR) (2017)
15. Hao, C., Yali, W., Guoyou, W., Yu, Q.: LSTD: A low-shot transfer detector for object detection. In: AAAI Conference on Artificial Intelligence (AAAI) (2018)
16. Hariharan, B., Girshick, R.B.: Low-shot visual recognition by shrinking and hallucinating features. In: IEEE International Conference on Computer Vision (ICCV) (2017)
17. He, K., Gkioxari, G., Dollár, P., Girshick, R.B.: Mask R-CNN. In: IEEE International Conference on Computer Vision (ICCV). pp. 2980–2988 (2017)
18. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) (2015)

19. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 770–778 (2016)
20. Hinterstoißer, S., Lepetit, V., Ilic, S., Holzer, S., Bradski, G.R., Konolige, K., Navab, N.: Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes. In: Asian Conference on Computer Vision (ACCV) (2012)
21. Hu, H., Gu, J., Zhang, Z., Dai, J., Wei, Y.: Relation networks for object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3588–3597 (2018)
22. Hu, S.X., Moreno, P., Xiao, Y., Shen, X., Obozinski, G., Lawrence, N., Damianou, A.: Empirical Bayes transductive meta-learning with synthetic gradients. In: International Conference on Learning Representations (ICLR) (2020)
23. Kang, B., Liu, Z., Wang, X., Yu, F., Feng, J., Darrell, T.: Few-shot object detection via feature reweighting. In: IEEE International Conference on Computer Vision (ICCV) (2019)
24. Kehl, W., Manhardt, F., Tombari, F., Ilic, S., Navab, N.: SSD-6D: Making RGB-based 3D detection and 6D pose estimation great again. In: IEEE International Conference on Computer Vision (ICCV). pp. 1530–1538 (2017)
25. Koch, G., Zemel, R., Salakhutdinov, R.: Siamese neural networks for one-shot image recognition. In: ICML workshops (2015)
26. Kuo, W., Angelova, A., Malik, J., Lin, T.Y.: ShapeMask: Learning to segment novel objects by refining shape priors. In: IEEE International Conference on Computer Vision (ICCV). pp. 9206–9215 (2019)
27. Lee, K., Maji, S., Ravichandran, A., Soatto, S.: Meta-learning with differentiable convex optimization. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
28. Li, F.F., Fergus, R., Perona, P.: One-shot learning of object categories. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) (2006)
29. Lin, T.Y., Dollár, P., Girshick, R.B., He, K., Hariharan, B., Belongie, S.J.: Feature pyramid networks for object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 936–944 (2017)
30. Lin, T.Y., Maire, M., Belongie, S.J., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common objects in context. In: European Conference on Computer Vision (ECCV) (2014)
31. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S.E., Fu, C.Y., Berg, A.C.: SSD: Single shot multibox detector. In: European Conference on Computer Vision (ECCV) (2016)
32. Massa, F., Russell, B.C., Aubry, M.: Deep exemplar 2D-3D detection by adapting from real to rendered views. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6024–6033 (2016)
33. Mousavian, A., Anguelov, D., Flynn, J., Kosecka, J.: 3D bounding box estimation using deep learning and geometry. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5632–5640 (2017)
34. Oberweger, M., Rad, M., Lepetit, V.: Making deep heatmaps robust to partial occlusions for 3D object pose estimation. In: European Conference on Computer Vision (ECCV) (2018)
35. Pavlakos, G., Zhou, X., Chan, A., Derpanis, K.G., Daniilidis, K.: 6-DoF object pose from semantic keypoints. In: IEEE International Conference on Robotics and Automation (ICRA). pp. 2011–2018 (2017)

36. Pitteri, G., Ilic, S., Lepetit, V.: CorNet: Generic 3D corners for 6D pose estimation of new objects without retraining. In: IEEE International Conference on Computer Vision Workshops (ICCVw) (2019)
37. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: PointNet: deep learning on point sets for 3D classification and segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 77–85 (2017)
38. Qi, H., Brown, M., Lowe, D.G.: Low-shot learning with imprinted weights. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5822–5830 (2017)
39. Rad, M., Lepetit, V.: BB8: A scalable, accurate, robust to partial occlusion method for predicting the 3D poses of challenging objects without using depth. In: IEEE International Conference on Computer Vision (ICCV). pp. 3848–3856 (2017)
40. Rahman, S., Khan, S.H., Porikli, F.M.: Zero-shot object detection: Learning to simultaneously recognize and localize novel concepts. In: Asian Conference on Computer Vision (ACCV) (2018)
41. Ravi, S., Larochelle, H.: Optimization as a model for few-shot learning. In: International Conference on Learning Representations (ICLR) (2017)
42. Redmon, J., Divvala, S.K., Girshick, R.B., Farhadi, A.: You only look once: Unified, real-time object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 779–788 (2016)
43. Redmon, J., Farhadi, A.: Yolo9000: Better, faster, stronger. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6517–6525 (2017)
44. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement (2018), arXiv preprint arXiv:1804.02767
45. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. In: International Conference on Neural Information Processing Systems (NeuIPS) (2015)
46. Schwartz, E., Karlinsky, L., Shtok, J., Harary, S., Marder, M., Pankanti, S., Feris, R.S., Kumar, A., Giryes, R., Bronstein, A.M.: RepMet: Representative-based metric learning for classification and few-shot object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5192–5201 (2019)
47. Shen, X., Efros, A.A., Aubry, M.: Discovering visual patterns in art collections with spatially-consistent feature learning. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
48. Snell, J., Swersky, K., Zemel, R.S.: Prototypical networks for few-shot learning. In: International Conference on Neural Information Processing Systems (NeurIPS) (2017)
49. Song, H.O., Lee, Y.J., Jegelka, S., Darrell, T.: Weakly-supervised discovery of visual pattern configurations. In: International Conference on Neural Information Processing Systems (NeurIPS) (2014)
50. Su, H., Qi, C.R., Li, Y., Guibas, L.J.: Render for CNN: Viewpoint estimation in images using CNNs trained with rendered 3D model views. In: IEEE International Conference on Computer Vision (ICCV). pp. 2686–2694 (2015)
51. Sundermeyer, M., Marton, Z.C., Durner, M., Brucker, M., Triebel, R.: Implicit 3D orientation learning for 6D object detection from RGB images. In: European Conference on Computer Vision (ECCV) (2018)
52. Tekin, B., Sinha, S.N., Fua, P.: Real-time seamless single shot 6D object pose prediction. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 292–301 (2018)

53. Tseng, H.Y., Mello, S.D., Tremblay, J., Liu, S., Birchfield, S., Yang, M.H., Kautz, J.: Few-shot viewpoint estimation. In: British Machine Vision Conference (BMVC) (2019)
54. Tulsiani, S., Carreira, J., Malik, J.: Pose induction for novel object categories. In: IEEE International Conference on Computer Vision (ICCV). pp. 64–72 (2015)
55. Tulsiani, S., Malik, J.: Viewpoints and keypoints. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
56. Vinyals, O., Blundell, C., Lillicrap, T.P., Kavukcuoglu, K., Wierstra, D.: Matching networks for one shot learning. In: International Conference on Neural Information Processing Systems (NeurIPS) (2016)
57. Wang, X., Huang, T.E., Darrell, T., Gonzalez, J.E., Yu, F.: Frustratingly simple few-shot object detection. In: International Conference on Machine Learning (ICML) (July 2020)
58. Wang, Y.X., Hebert, M.: Learning to learn: Model regression networks for easy small sample learning. In: European Conference on Computer Vision (ECCV) (2016)
59. Wang, Y.X., Ramanan, D., Hebert, M.: Meta-learning to detect rare objects. In: IEEE International Conference on Computer Vision (ICCV) (October 2019)
60. Xiang, Y., Kim, W., Chen, W., Ji, J., Choy, C., Su, H., Mottaghi, R., Guibas, L., Savarese, S.: ObjectNet3D: A large scale database for 3D object recognition. In: European Conference on Computer Vision (ECCV) (2016)
61. Xiang, Y., Mottaghi, R., Savarese, S.: Beyond PASCAL: A benchmark for 3D object detection in the wild. In: IEEE Winter Conference on Applications of Computer Vision (WACV) (2014)
62. Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes. In: Robotics: Science and Systems (RSS) (2018)
63. Xiao, Y., Qiu, X., Langlois, P., Aubry, M., Marlet, R.: Pose from shape: Deep pose estimation for arbitrary 3D objects. In: British Machine Vision Conference (BMVC) (2019)
64. Yan, X., Chen, Z., Xu, A., Wang, X., Liang, X., Lin, L.: Meta R-CNN : Towards general solver for instance-level low-shot learning. In: IEEE International Conference on Computer Vision (ICCV) (2019)
65. Zhou, X., Karpur, A., Luo, L., Huang, Q.: StarMap for category-agnostic keypoint and viewpoint estimation. In: European Conference on Computer Vision (ECCV) (2018)
66. Zhu, P., Wang, H., Saligrama, V.: Zero shot detection. IEEE Transactions on Circuits and Systems for Video Technology (TCSVT) (2019)