Multi-Faceted Distillation of Base-Novel Commonality for Few-shot Object Detection

Shuang Wu[2], Wenjie Pei[2,*], Dianwen Mei[2], Fanglin Chen[2], Jiandong Tian[3], and Guangming Lu[1,2,*]

[1] Guangdong Provincial Key Laboratory of Novel Security Intelligence Technologies
[2] Harbin Institute of Technology, Shenzhen, China
[3] Shenyang Institute of Automation, Chinese Academy of Sciences
{wushuang9811, wenjiecoder}@outlook.com, {178mdw, linwers}@gmail.com, luguangm@hit.edu.cn, tianjd@sia.cn

Abstract. Most of existing methods for few-shot object detection follow the fine-tuning paradigm, which potentially assumes that the class-agnostic generalizable knowledge can be learned and transferred implicitly from base classes with abundant samples to novel classes with limited samples via such a two-stage training strategy. However, it is not necessarily true since the object detector can hardly distinguish between class-agnostic knowledge and class-specific knowledge automatically without explicit modeling. In this work we propose to learn three types of class-agnostic commonalities between base and novel classes explicitly: recognition-related semantic commonalities, localization-related semantic commonalities and distribution commonalities. We design a unified distillation framework based on a memory bank, which is able to perform distillation of all three types of commonalities jointly and efficiently. Extensive experiments demonstrate that our method can be readily integrated into most of existing fine-tuning based methods and consistently improve the performance by a large margin.

Keywords: Few-shot; Object Detection; Knowledge Distillation; Commonality

1 Introduction

Few-shot object detection aims to learn effective object detectors for novel classes with limited samples, leveraging the generalizable prior knowledge learned from abundant data of base classes. Compared to general object detection [8,27], few-shot object detection is supposed to be able to generalize across different classes rather than just across different samples within a class. It is also more challenging than few-shot classification [7,31,34] in that it demands to learn the transferable knowledge not only on recognition, but also on localization.

A prominent modeling paradigm for few-shot object detection is fine-tuning framework [6,26,32,35,40], which first pre-trains the object detector using the

* Corresponding authors.

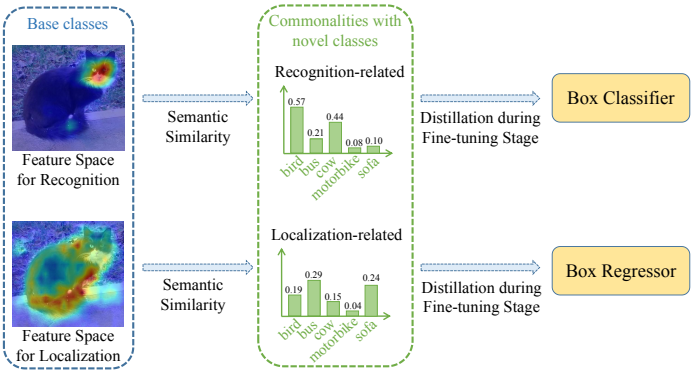

Fig. 1. Given a cat sample from the base class 'Cat', we measure the semantic similarities between it and each of novel classes in the optimized feature space for both object recognition and localization, which are interpreted as the recognition- and localization-related semantic commonalities, respectively. These learned commonalities are distilled during the fine-tuning stage to improve the performance of the object detector on novel classes. Note that the visualizations by Grad-CAM++ [2] show that the learned features for recognition focus on the local salient regions while the localization pays more attention to the global boundary or shape features.

samples from base classes, then fine-tunes the model on novel classes. Based on such two-stage training strategy, many methods are proposed to deal with a specific challenge in few-shot object detection, such as MPSR [40] which tackles the problem of scale variation, FSCE [32] for alleviating confusion between novel classes, and Retentive R-CNN [6] suppressing the performance degradation on base classes during fine-tuning. A potential hypothesis of such fine-tuning paradigm is that the class-agnostic prior knowledge for object detection could be transferred from base classes to novel classes implicitly. Nevertheless, the object detector can hardly distinguish between class-agnostic knowledge and class-specific knowledge automatically without explicit modeling.

In this work we propose to learn multi-faceted commonalities between base classes and novel classes explicitly in the fine-tuning framework, which is class-agnostic and can be transferred across different classes. Then we perform distillation on the learned commonalities to circumvent the scarcity of novel classes and thereby improve the performance of the object detector on novel classes. To be specific, we aim to learn three types of base-novel commonalities: 1) the recognition-related semantic commonalities like similar appearance features shared among semantically close classes; 2) the localization-related semantic commonalities such as the similar object shape or boundary features between different classes; 3) the distribution commonalities in feature space shared between similar classes like close mean and variance of features in a presumed Gaussian distribution [30]. Consider the example in Figure 1, we first learn the optimized feature spaces for object recognition and localization respectively. Then we measure the semantic similarities between a given cat sample (from the base class

'Cat') and each of novel classes in each feature space. The obtained similarity distribution in the feature space for recognition is interpreted as the recognition-related semantic commonalities, and the same applies to the localization-related semantic commonalities. The learned commonalities are further distilled towards their corresponding tasks respectively during fine-tuning of the object detector on novel classes, namely recognition-related commonalities for object classification and localization-related commonalities for object bounding box regression. Consequently, all samples in base classes that share commonalities with a novel class can be leveraged to train the object detector on this novel class, which is equivalent to augment the training data for novel classes. Note that the learned features for recognition and localization focus on different object areas: the recognition captures the local salient regions (e.g., the head of cat in Figure 1) whilst the localization pays more attention to the global boundaries as shown in Figure 1. Thus we decouple the feature spaces for object recognition and localization and learn the corresponding commonalities in the decoupled feature spaces separately. Inspired by Distribution Calibration [45], we learn the distribution commonalities by estimating the feature variance for a novel class via reference to the closed base classes, and distill the obtained commonalities by sampling for data augmentation. To conclude, we make following contributions.

- We learn three types of generalizable commonalities between base and novel classes explicitly, which can be transferred from base classes to novel classes.
- We design a unified distillation framework based on a memory bank, which is able to distill all three types of learned commonalities jointly and efficiently in an end-to-end manner during the fine-tuning stage.
- Our method can be integrated into most of fine-tuning based methods. Extensive experiments show that our method leads to substantial improvements when integrated into various classical methods. As a result, our method advances the state-of-the-art performance by a large margin.

2 Related Work

Few-Shot Image Classification. Few-shot image classification, which aims to recognize novel categories with limited annotated instances, has received increased attention in the recent past. Optimization-based approaches [7,20,24] modify the classical gradient-based optimization for fast adaption to new tasks. Metric-based approaches [31,33,34,48] learn a metric space where instances could be recognized by comparing the distance to the prototype of each category. Hallucination-based approaches [11,36,45] learn to generate novel samples to deal with data scarcity. Compared to image classification, few-shot object detection which has to consider localization in addition, is still under-explored.

Few-Shot Object Detection. Early works of few-shot object detection focus on the meta-learning paradigm [5,10,15,16,21,22,37,42,44,50], which introduces a meta-learner to leverage meta-level knowledge that can be transferred from base classes to novel classes. Recently, researchers find out that the simple fine-tuning based approaches [1,6,23,26,32,35,38,39,40,51,52] could outperform most

of meta-learning based approaches. TFA [35] proposes a two-stage fine-tuning process that only fine-tunes the prediction layer. FSCE [32] rescues misclassifications between novel classes by supervised contrastive learning. UP-FSOD [38] devises universal prototypes to enhance the generalization of object features. Retentive R-CNN [6] regularizes the adaptation during fine-tuning to maintain the performance on base classes. DeFRCN [26] proposes to decouple the features for RPN and R-CNN. All these methods learn to detect novel instances by implicitly exploiting the class-agnostic knowledge learned from base classes. Instead, we address few-shot object detection by distilling the multi-faceted commonalities between base classes and novel classes.

Knowledge Distillation. Classical knowledge distillation aims at transferring knowledge from a model (teacher) to the other (student). [14] introduces the soft prediction of the teacher network as dark knowledge for distillation. [28] leverages the intermediate representations learned by teacher to guide student. [19] proposes to transfer attention information of teacher. Several works [9,18,43,47,49] use the student itself as a teacher, named self-distillation. Inspired by these works, we design a novel distillation framework to distill commonalities between base classes and novel classes based on a memory bank.

3 Multi-Faceted Distillation of Base-Novel Commonality

In this section, we start with the preliminary of few-shot object detection, then we introduce our method which distills the multi-faceted base-novel commonalities to circumvent the scarcity of training samples in few-shot object detection.

3.1 Preliminary

We follow the standard few-shot object detection settings introduced in [16,35] and split classes into two sets: base classes C_b with abundant annotated instances, and novel classes C_n with only K (usually less than 30) instances per category. Our proposed method involves the two-stage training procedure [35]. In the first stage, the Faster R-CNN [27] detector is trained with all the available samples of base classes. In the second stage, the pre-trained detector is fine-tuned on samples of both base and novel classes.

Different from existing works [6,26,32,35,40] that create a small balanced training set with K novel samples and K base samples in the second stage, we fine-tune the detector with abundant samples of base classes which are used in the first stage (the training details are described in the supplementary materials). Thus, we are able to distill the multi-faceted commonalities that can be transferred from abundant samples of base classes to limited samples of novel classes to circumvent the data scarcity. Specifically, we distill three types of base-novel commonalities to learn robust detector for novel classes, including 1) the recognition-related semantic commonalities 2) the localization-related semantic commonalities, and 3) the distribution commonalities. Figure 2 illustrates the overall framework of our method.

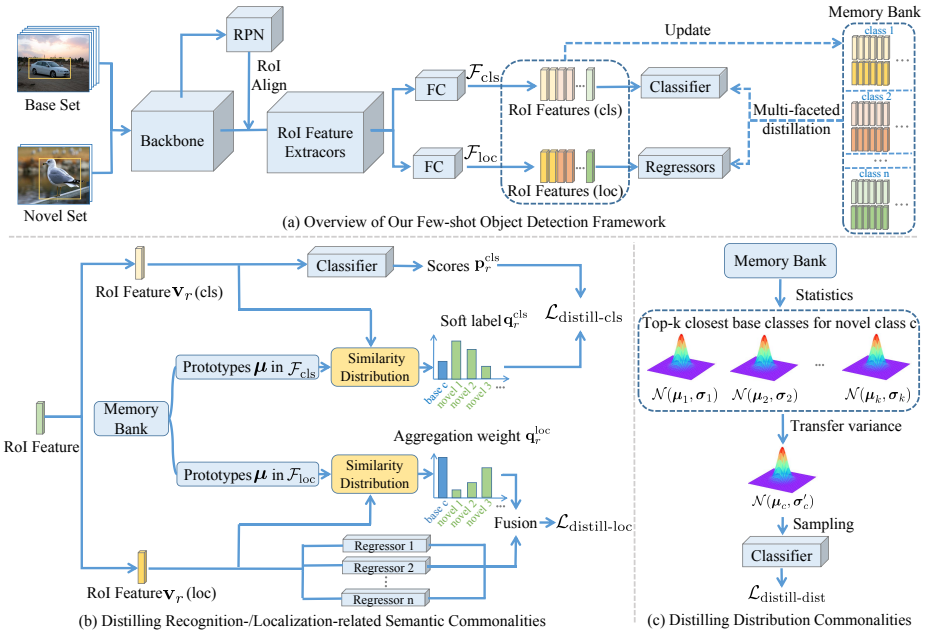

Fig. 2. The framework of our approach. (a) The RoI features are decoupled into two separate feature spaces for classification $\mathcal{F}_{\text{cls}}$ and bounding box regression $\mathcal{F}_{\text{loc}}$, respectively. During the fine-tuning stage, the recognition-related and distribution commonalities are learned in $\mathcal{F}_{\text{cls}}$ while the localization-related commonalities are learned in $\mathcal{F}_{\text{loc}}$. All three types of commonalities are distilled in a unified framework based on a memory bank. (b) The recognition-related commonalities are distilled by viewing them as the soft labels to supervise the classifier whereas the localization-related commonalities are used as aggregation weights to fuse all regressors. (c) We distill the variance for a novel class via reference to the top-k closest base classes, and sample examples from the calibrated distribution to train the classifier.

3.2 Distilling Recognition-related Semantic Commonalities

Semantically close categories tend to share similar high-level semantic commonalities that is related to object recognition, such as similar appearance between cow and horse. We aim to distill such semantic commonalities between base and novel classes to guide the learning of the object detector on novel classes.

Classical knowledge distillation [14] transfers knowledge from a larger teacher model to a student model. The transferred knowledge is represented as the predicted probabilistic distribution on all classes by the teacher model, which can be interpreted as the similarities of current sample to each class. The knowledge distillation is performed by using such probabilistic distribution as the soft labels to supervise the learning of the model together with the one-hot hard labels.

We draw inspiration from such classical way of knowledge distillation but conduct distillation in a different way. To distill the recognition-related semantic commonalities between base and novel classes, we measure the similarities of samples in base classes to each novel class. Since there is no sufficient samples from novel classes for learning a teacher model, we calculate such similarities in a pre-learned feature space $\mathcal{F}_{\text{cls}}$ directly instead of predicting class probabilities by a teacher model. Formally, given a foreground region proposal r from a base class, which is generated by the region proposal network (RPN), we define the similarity of it to a novel class c as the cosine distance between its RoI feature $\mathbf{v}_r$ and the prototype $\boldsymbol{\mu}_c$ of the class c in the pre-learned feature space $\mathcal{F}_{\text{cls}}$:

$$d_r^c = \alpha \cdot \frac{\mathbf{v}_r^T \boldsymbol{\mu}_c}{\|\mathbf{v}_r\| \, \|\boldsymbol{\mu}_c\|}, c \in \mathcal{C}_n. \tag{1}$$

Herein, $\mathcal{C}_n$ is the set of novel classes and $\alpha > 0$ is the scaling factor. The prototype $\boldsymbol{\mu}_c$ is obtained by averaging the object features of a candidate set (implemented as a memory bank, will be elaborated on in Section 3.5) in the novel class c:

$$\boldsymbol{\mu}_c = \frac{1}{n_c} \sum_{i=1}^{n_c} \mathbf{f}_c^i, \tag{2}$$

where $\mathbf{f}_c^i$ is the vectorial feature for the i-th object in the candidate set and n_c is size of the set. Since we focus on distilling the base-novel commonalities to circumvent the scarcity of training samples in novel classes, the base-base commonalities are ignored to allocate all model capacity to base-novel commonalities. As a result, the similarities of a region proposal r from a base class to other base classes are defined as a small constant value:

$$d_r^c = -\alpha, c \in \mathcal{C}_b \setminus \{c_{\text{gt}}\}, \tag{3}$$

where $\mathcal{C}_b$ denotes the set of base classes and α is the same scaling factor as in Equation 1. Note that we also calculate the cosine similarity between r and its groundtruth class c_{gt} following Equation 1 to guarantee the predicting accuracy (w.r.t. c_{gt}). Finally we normalize the similarities of sample r to all classes by a softmax function:

$$\mathbf{q}_{r,c}^{\text{cls}} = \frac{\exp(d_r^c)}{\sum_{i=1}^{C} \exp(d_r^i)}, c \in \mathcal{C}_n \cup \mathcal{C}_b. \tag{4}$$

Assuming that a foreground region proposal r has 0 commonality with background c_{bg}, we obtain the complete similarity distribution for r: $\mathbf{q}_r^{\text{cls}} = [\mathbf{q}_r^{\text{cls}}; 0]$.

Similar to the classical knowledge distillation, we utilize the obtained similarities of a region proposal as soft labels to supervise the learning of our object detector. In particular, we perform such distillation during the fine-tuning stage of the detector. Formally, for the region proposal r from a base class, we minimize the Kullback-Leibler (KL) divergence between the soft labels $\mathbf{q}_r^{\text{cls}}$ and the predicted class probabilities $\mathbf{p}_r^{\text{cls}}$ by the object detector:

$$\mathcal{L}_{\text{distill-cls}} = \sum_{c \in \mathcal{C}_n \cup \mathcal{C}_b \cup \{c_{\text{bg}}\}} (\mathbf{q}_{r,c}^{\text{cls}} \log \mathbf{q}_{r,c}^{\text{cls}} - \mathbf{q}_{r,c}^{\text{cls}} \log \mathbf{p}_{r,c}^{\text{cls}}). \tag{5}$$

Rationale. We learn the semantic commonalities that are related to object recognition by measuring the similarities of samples from base classes to each novel class in a pre-defined feature space. Then the learned commonalities (after normalization) are viewed as soft labels to supervise the fine-tuning of the object detector. Consequently, all samples in base classes that share recognition-related semantics with a novel class can be leveraged to train the object detector on this novel class. In this sense, the proposed commonality distillation significantly augments the training data for novel classes, thereby improving the performance of the object detector on novel classes.

3.3 Distilling Localization-related Semantic Commonalities

Besides the recognition-related semantic commonalities, similar categories also share semantic commonalities that is related to object localization such as similar shape or boundary features. Distilling such commonalities between similar base and novel classes enables the object detector to learn transferable knowledge on localization from abundant base class samples, thereby improving its performance of object detection on novel classes.

The localization-related semantic commonalities are distilled in a similar way as the recognition-related commonalities in Section 3.2. We also learn the localization-related commonalities by measuring the similarities of samples in base classes to each novel class in a pre-learned feature space $\mathcal{F}_{\text{loc}}$. One of the key differences between distillation of two different types of commonalities (recognition- or localization-related) is that they are learned in different pre-learned feature spaces: each feature space should be learned by optimizing the corresponding task (object classification or localization), as illustrated in Figure 1. We present an efficient implementation in Section 3.5.

The learned localization-related commonalities is represented as the normalized similarities in the same form shown in Equation 4. In contrast to viewing the recognition-related commonalities as soft labels for supervision, the localization-related commonalities are leveraged as normalized weights to aggregate all class-specific bounding box regressors for object localization. This is based on the intuition that an object can be localized by not only the bounding box regressor for its groundtruth class, but also the regressors for the similar classes, more similarities leading to more confidence. Formally, given a region proposal r from a base class, its bounding box is predicted as offsets $\mathbf{t} = (t_x, t_y, t_w, t_h)$ to the groundtruth position by aggregating the predictions of all regressors for C classes. Then the detector is optimized by minimizing the error between the aggregated prediction and the groundtruth using the smoothed L1 loss [8]:

$$\mathcal{L}_{\text{distill-loc}} = \sum_{c=1}^{C} \mathbf{q}_{r,c}^{\text{loc}} \cdot \sum_{i \in \{x,y,w,h\}} \text{Smooth}_{L1}(t_i^c - u_i), \tag{6}$$

where $\mathbf{q}_r^{\text{loc}}$ is the normalized similarities representing the localization-related commonalities. u_i is the bounding-box regression groundtruth for r while t_i^c is the prediction of box regressor for the class c.

Rationale. The similarities between samples from base classes to each novel class in a pre-learned feature space $\mathcal{F}_{\text{loc}}$ towards localization are learned as the localization-related commonalities, and are further used as aggregation weights to fuse regressors for all classes. To be specific, a sample (object) from a base class is localized by referring to the predictions of all regressors for the novel classes sharing localization-related commonalities with this sample. It is equivalent to training these regressors with the sample. As a result, all regressors for novel classes are optimized with a lot of additional training samples from base classes, which yields better performance of localization.

3.4 Distilling Distribution Commonalities

Semantically similar categories usually follow similar data distributions, such as close mean and variance of features in a presumed Gaussian distribution between these categories [30]. Hence, the third type of commonalities between base and novel classes that we aim to distill is the distribution commonalities. Inspired by Distribution Calibration [45] in few-shot image classification, we distill the distributional statistics from base classes to calibrate the distribution of those similar novel classes. Consequently, we can sample sufficient examples for these novel classes to improve the performance of the object detector on novel classes.

Unlike Distribution Calibration which transfers both the mean and variance of base classes to novel classes, we only distill the variance of base classes while preserving the mean values of novel classes. This is because transferring both the mean and variance of base classes would result in the distributional overlapping between the base and novel classes, making it harder to distinguish between them during object detection. In contrast, the classification between base and novel classes is not required in the few-shot classification setting.

Assuming that each feature dimension follows a Gaussian distribution, which is consistent with Distribution Calibration [45], we first calculate the mean and variance per feature dimension for both base and novel classes in a pre-learned feature space, and select the top-k semantically closest base classes for each novel class according to the Euclidean distance w.r.t. the mean values (equivalent to the class prototype in Equation 2). Then we can approximate the variance of a novel class using the averaged variance over its top-k closest base classes. Formally, the calibrated variance of a novel class c is estimated by:

$$\boldsymbol{\sigma}'_c = \frac{1}{k} \sum_{i \in S_c} \boldsymbol{\sigma}_i. \tag{7}$$

Herein, $\boldsymbol{\sigma}_i$ is the variance of the base class i and S_c is the set of top-k closest base classes to the novel class c. In this way we are able to sample more examples in this pre-learned feature space for the novel class c following the obtained Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_c, \boldsymbol{\sigma}'_c)$:

$$\mathbb{S}_c = \{\mathbf{v} | \mathbf{v} \sim \mathcal{N}(\boldsymbol{\mu}_c, \boldsymbol{\sigma}'_c)\}, \tag{8}$$

where $\boldsymbol{\mu}_c$ is mean of the novel class c. $\mathbb{S}_c$ is the set of sampled features, which are further used to train the classifier f_θ of the object detector using the Cross-Entropy loss:

$$\mathcal{L}_{\text{distill-dist}} = \frac{1}{|\mathbb{S}_c|} \sum_{\mathbf{v} \in \mathbb{S}_c} \text{CE}(c, f_\theta(\mathbf{v})). \tag{9}$$

3.5 Unified Distillation Framework Based on Memory Bank

We propose a unified distillation framework, which is able to distill all three commonalities jointly in an end-to-end manner during the fine-tuning stage.

Both the recognition-related commonalities and the localization-related commonalities are obtained by calculating the similarities between samples of base classes to each of novel classes in their corresponding (but different) pre-learned feature spaces. Typically such pre-learned feature spaces are independent from the feature space for learning the detector, which is achieved by pre-learning the feature spaces based on other data or other networks. Doing so enables the knowledge distillation between two different feature spaces. However, such implementation has two limitations: 1) the commonalities calculated in the pre-learned feature space may not be accurate since the extracted features for samples of both base and novel classes are potentially not optimized; 2) the whole training is performed in two separated stages, which is not efficient.

We propose to learn the commonalities in the same feature space as that for learning the detector. As shown in Figure 2, we only learn one feature space by the typical feature learning backbone together with the RoI feature extractor based on the training data for current task. Then we decouple the feature space into two separate feature spaces by two projection heads: one (denoted as $\mathcal{F}_{\text{cls}}$) is connected to the classification head and is used for learning the recognition-related commonalities, the other one (denoted as $\mathcal{F}_{\text{loc}}$) is connected to the regression head and is used for learning the localization-related commonalities. Each projection head consists of a fully connected layer and a ReLU layer. We first pre-train the detector based on the samples from base classes. Then in the fine-tuning stage, we learn each type of commonalities and perform the commonality distillation jointly in the corresponding feature space. Note that the distribution commonalities are also learned in the feature space $\mathcal{F}_{\text{cls}}$ since the distribution similarities are intuitively more related to the recognition-related semantics.

Commonality Distillation. During the fine-tuning of the detector, the feature space is evolving all the time. Thus all types of commonalities are also evolving with the update of the feature space. Meanwhile, the commonality distillation is performed in two aspects. First, the commonalities learned based on the previous training state of feature space are further used to optimize the feature space in the next iteration (state). In this sense, the commonalities are distilled between different training states in the same feature space, which is similar to Self-Knowledge Distillation [18]. Second, the recognition-related and distribution commonalities are also distilled from the feature space $\mathcal{F}_{\text{cls}}$ to the classification head while the localization-related commonalities are distilled from $\mathcal{F}_{\text{loc}}$ to the localization head, yielding more precise classifier and regressors.

Memory Bank. During the fine-tuning of the detector, the commonalities are evolving with the update of the feature space. However, calculating the prototype for each class (including base and novel classes) from scratch using all available samples in the training set, which is involved in learning all three types of commonalities, is quite computationally expensive due to the feature extraction for all samples. To address this problem, we maintain a dynamic memory bank to store the features (in both $\mathcal{F}_{\text{loc}}$ and $\mathcal{F}_{\text{cls}}$) of a maximum number of L RoI features for each class to improve the efficiency. Denoting the memory bank as $\mathbf{M} = \{\mathbf{m}_c\}_{c=1}^{C}$ where C is the class number, the RoI features of each class are stored as a queue. During each training iteration, we update the memory bank by enqueuing the current batch of samples to the corresponding class queue and dequeuing the same amount of oldest samples for the same class. Then we can calculate the prototype for each class using the RoI features stored in $\mathbf{M}$. As a result, we do not need to extract features for all samples from scratch each time the feature space is updated, and the operating efficiency is thereby improved significantly. Using memory bank for efficiency has been previously explored in unsupervised learning [12,41].

Parameter Learning. In the pre-training stage using samples from base classes, we train the object detector with standard Faster R-CNN [27] losses:

$$\mathcal{L}_{\text{det}} = \mathcal{L}_{\text{rpn}} + \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{reg}}, \tag{10}$$

where $\mathcal{L}_{\text{rpn}}$ is the loss of the RPN to distinguish foreground from background, $\mathcal{L}_{\text{cls}}$ is the Cross-Entropy loss for classification, and $\mathcal{L}_{\text{reg}}$ is the smoothed L1 loss [8] for the regression of bounding boxes. In the fine-tuning stage, the model is supervised with both the Faster R-CNN loss $\mathcal{L}_{\text{det}}$ and the losses for the distillation of three types of commonalities, in an end-to-end manner:

$$\mathcal{L} = \mathcal{L}_{\text{det}} + \lambda_c \mathcal{L}_{\text{distill-cls}} + \lambda_l \mathcal{L}_{\text{distill-loc}} + \lambda_d \mathcal{L}_{\text{distill-dist}}, \tag{11}$$

where λ_c, λ_l and λ_d are hyper-parameters to balance among losses.

4 Experiments

4.1 Experimental Setup

Datasets. Our approach is evaluated on PASCAL VOC [4] and MS COCO [25] datasets. We follow the consistent data construction and evaluation protocol in [16,35]. For PASCAL VOC, the overall 20 classes are split into 15 base classes and 5 novel classes. We utilize the same three partitions of base classes and novel classes introduced in [16]. All base class instances from PASCAL VOC (07+12) trainval sets are available. Each novel class has K instances available where K is set to 1, 2, 3, 5 and 10. We report AP50 of novel classes (nAP50) on PASCAL VOC 07 test set. For the 80 classes in MS COCO, the 20 classes overlapped with PASCAL VOC are selected as novel classes, the remaining 60 classes are selected as base classes. Similarly, we report COCO-style AP and AP75 of novel classes on COCO 2014 validation set with $K = 1, 2, 3, 5, 10, 30$.

Table 1. Comparison of different few-shot object detection methods in terms of nAP50 on three PASCAL VOC Novel Split sets.

Method / Shots	Novel Split 1					Novel Split 2					Novel Split 3				
	1	2	3	5	10	1	2	3	5	10	1	2	3	5	10
LSTD [3]	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
FSRW [16]	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 [37]	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 [44]	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
RepMet [17]	26.1	32.9	34.4	38.6	41.3	17.2	22.1	23.4	28.3	35.8	27.5	31.1	31.5	34.4	37.2
NP-RepMet [46]	37.8	40.3	41.7	47.3	49.4	41.6	43.0	43.4	47.4	49.1	33.3	38.0	39.8	41.5	44.8
TFA w/cos [35]	39.8	36.1	44.7	55.7	56.0	23.5	26.9	34.1	35.1	39.1	30.8	34.8	42.8	49.5	49.8
MPSR [40]	41.7	–	51.4	55.2	61.8	24.4	–	39.2	39.9	47.8	35.6	–	42.3	48.0	49.7
HallucFsDet [51]	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
Retentive R-CNN[6]	42.4	45.8	45.9	53.7	56.1	21.7	27.8	35.2	37.0	40.3	30.2	37.6	43.0	49.7	50.1
FSCE [32]	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
FSCN [23]	40.7	45.1	46.5	57.4	62.4	27.3	31.4	40.8	42.7	46.3	31.2	36.4	43.7	50.1	55.6
SRR-FSD [52]	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
SQMG [50]	48.6	51.1	52.0	53.7	54.3	41.6	45.4	45.8	46.3	48.0	46.1	51.7	52.6	54.1	55.0
CME [22]	41.5	47.5	50.4	58.2	60.9	27.2	30.2	41.4	42.5	46.8	34.3	39.6	45.1	48.3	51.5
Dictionary [39]	46.1	43.5	48.9	60.0	61.7	25.6	29.9	44.8	47.5	48.2	39.5	45.4	48.9	53.9	56.9
FADI [1]	50.3	54.8	54.2	59.3	63.2	30.6	35.0	40.3	42.8	48.0	45.7	49.7	49.1	55.0	59.6
UP-FSOD [38]	43.8	47.8	50.3	55.4	61.7	31.2	30.5	41.2	42.2	48.3	35.5	39.7	43.9	50.6	53.3
QA-FewDet [10]	42.4	51.9	55.7	62.6	63.4	25.9	37.8	46.6	48.9	51.1	35.2	42.9	47.8	54.8	53.5
DeFRCN [26]	57.0	58.6	64.3	67.8	67.0	35.8	42.7	51.0	54.5	52.9	52.5	56.6	55.8	60.7	62.5
Ours	**63.4**	**66.3**	**67.7**	**69.4**	**68.1**	**42.1**	**46.5**	**53.4**	**55.3**	**53.8**	**56.1**	**58.3**	**59.0**	**62.2**	**63.7**

Implementation Details. As a plug-and-play module, our approach can be easily integrated into other fine-tuning based methods. we evaluate our approach on four baselines: TFA [35], Retentive R-CNN [6], FSCE [32] and DeFRCN [26]. We train the detector with a mini-batch of 16 on 8 GPUs, 2 images per GPU. ResNet-101 [13] pre-trained on ImageNet [29] is used as the backbone. The maximum queue size L in our memory bank is tuned to be 2048. The scaling factor α is tune to be 5. For distribution distillation, we transfer the variance of top $k = 2$ base classes, and sample $|\mathbb{S}_c| = 10$ instances from the calibrated distribution for novel class c during each iteration. The weights of each loss are tuned to be $\lambda_c = 0.1$, $\lambda_l = 1.0$, $\lambda_d = 0.1$. Moreover, We begin the distillation after 200 iterations in the fine-tuning stage to perform a basic optimization of the feature space on novel classes. Code is available at: `https://github.com/WuShuang1998/MFDC`.

4.2 Comparison with State-of-the-art Methods

We integrate our method based on DeFRCN [26], a state-of-the-art method for few-shot object detection, to compare with other latest methods.

Results on PASCAL VOC. Table 1 shows the results on PASCAL VOC. It can be observed that our approach outperforms other methods in all novel splits with different numbers of training shots. In particular, our method achieves much larger performance gain in extremely low-shot settings. For instance, for novel split 1, our approach surpasses the previously best method by 6.4% and 7.7% in 1-shot and 2-shot scenarios, respectively. It is reasonable because the distillation of commonalities plays more important role in fewer-shot settings.

Table 2. Few-shot object detection performance on MS COCO.

Method	1-shot		2-shot		3-shot		5-shot		10-shot		30-shot	
	nAP	nAP75	nAP	nAP75	nAP	nAP75	nAP	nAP75	nAP	nAP75	nAP	nAP75
FSRW [16]	–	–	–	–	–	–	–	–	5.6	4.6	9.1	7.6
SRR-FSD [52]	–	–	–	–	–	–	–	–	11.3	9.8	14.7	13.5
FSCE [32]	–	–	–	–	–	–	–	–	11.9	10.5	16.4	16.2
UP-FSOD [38]	–	–	–	–	–	–	–	–	11.0	10.7	15.6	15.7
SQMG [50]	–	–	–	–	–	–	–	–	13.9	11.7	15.9	14.3
CME [22]	–	–	–	–	–	–	–	–	15.1	16.4	16.9	17.8
TFA w/cos [35]	3.4	3.8	4.6	4.8	6.6	6.5	8.3	8.0	10.0	9.3	13.7	13.4
MPSR [40]	2.3	2.3	3.5	3.4	5.2	5.1	6.7	6.4	9.8	9.7	14.1	14.2
QA-FewDet [10]	4.9	4.4	7.6	6.2	8.4	7.3	9.7	8.6	11.6	9.8	16.5	15.5
FADI [1]	5.7	6.0	7.0	7.0	8.6	8.3	10.1	9.7	12.2	11.9	16.1	15.8
DeFRCN [26]	6.5	6.9	11.8	12.4	13.4	13.6	15.3	14.6	18.6	17.6	22.5	22.3
Ours	**10.8**	**11.6**	**13.9**	**14.8**	**15.0**	**15.5**	**16.4**	**17.3**	**19.4**	**20.2**	**22.7**	**23.2**

Table 3. Performance of integrating our method with different classical methods in term of nAP50 on Novel Split 1 of PASCAL VOC.

Baseline Method	Ours	1-shot	2-shot	3-shot	5-shot	10-shot
TFA w/cos [35]		39.8	36.1	44.7	55.7	56.0
	✓	**45.2**	**47.3**	**50.6**	**58.2**	**58.4**
Retentive R-CNN [6]		42.4	45.8	45.9	53.7	56.1
	✓	**47.8**	**48.1**	**51.4**	**58.2**	**58.9**
FSCE [32]		44.2	43.8	51.4	61.9	63.4
	✓	**48.0**	**51.6**	**55.3**	**63.8**	**66.2**
DeFRCN [26]		57.0	58.6	64.3	67.8	67.0
	✓	**63.4**	**66.3**	**67.7**	**69.4**	**68.1**

Results on MS COCO. Similar performance improvements by our method can be observed on the MS COCO benchmark. As shown in Table 2, our approach consistently outperforms other state-of-the-art methods in all settings although MS COCO is quite challenging. Particularly, for 1-shot scenarios, our approach pushes forward the current state-of-the-art performance from 6.5% to 10.8% in nAP. Besides, the improvement from 6.9% to 11.6% in nAP75 demonstrates the effectiveness of our approach on localization.

4.3 Integration with Different Baseline Methods

We further integrate our method with different baseline methods to evaluate the robustness of our method. Table 3 presents the performance of four different baselines and our method on Novel Split 1 of PASCAL VOC. Our method consistently boosts the performance distinctly. For instance, when integrated with TFA w/cos [35], our method achieves substantial performance gains: 5.4%, 11.2%, 5.9%, 2.5% and 2.4% from 1-shot to 10-shot respectively. These results reveal the strong robustness of our approach on different baseline methods.

Table 4. Effectiveness of each type of commonality. 'Recog', 'Local', 'Dist' refer to the recognition-related, localization-related and distribution commonalities, respectively.

Recog	Local	Dist	nAP50 1-shot	nAP50 2-shot	nAP50 3-shot
			58.5	62.6	65.4
✓			62.3	64.8	67.3
	✓		59.9	64.1	65.7
		✓	62.6	65.1	66.2
✓		✓	63.2	65.9	67.7
✓	✓		62.8	65.6	67.2
✓	✓	✓	**63.4**	**66.3**	**67.7**

Table 5. Effect of using different feature spaces from the object detector to learn commonalities. 'Independent' denotes the feature space pre-optimized on ImageNet, and 'uniform' denotes the same feature space as the object detector.

Feature space	nAP50 1-shot	nAP50 2-shot	nAP50 3-shot
Baseline	58.5	62.6	65.4
Independent	59.6	63.8	66.1
Uniform (ours)	**62.3**	**64.8**	**67.3**

4.4 Ablation Studies

In this section, we conduct ablation studies by integrating our method with DeFRCN [26]. All experiments are performed on Novel Split 1 of PASCAL VOC. Note that more ablation studies on other hyper-parameters are provided in the supplementary materials.

Effectiveness of each type of commonality. Table 4 shows the effectiveness of each type of commonality. Compared with the baseline in the first line, each individual type of commonality improves the performance distinctly. Combining all three types of commonalities achieves larger performance gain than any individual one.

Learning commonalities in an independent feature space from the object detector. Our method learns commonalities in the same (uniform) feature space as the object detector, which allows our model to 1) achieve more accurate commonalities due to more optimized features for current data and 2) perform commonality distillation in an end-to-end manner. To validate the first merit, we conduct experiments to learn commonalities in an independent feature space from the object detector, which is pre-optimized on ImageNet dataset. All class prototypes and cosine similarities for learning commonalities are calculated in this independent feature space. The results in Table 5 show that the performance gain in such way is smaller than that of using the same space feature as the object detector (denoted as 'Uniform').

Qualitative evaluation. To have a qualitative evaluation, we visualize the instances from base classes that have most recognition- and localization-related commonalities (interpreted as semantic similarities) with the novel class 'Bird' respectively in Figure 3(a). The instances from the semantically similar base classes to 'Bird', such as 'Dog' and 'sheep', tend to have more recognition-related commonalities with 'Bird' than other base classes. In contrast, instances from the base classes bearing more shape similarities to 'Bird', like 'Plane', have more localization-related commonalities with 'Bird' than other classes. Such observations are consistent with the different attention distributions in feature space between recognition and localization shown in Figure 1. By distilling the multi-

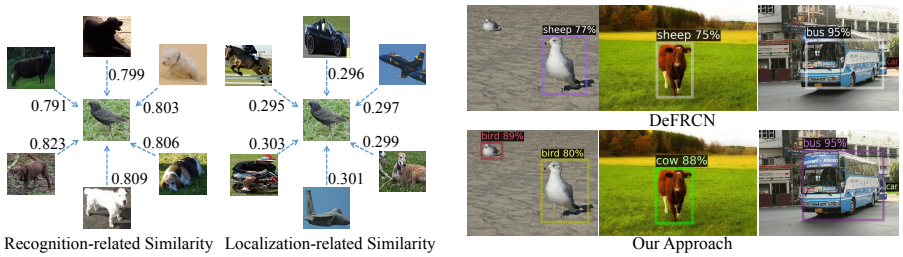

Fig. 3. (a) Visualization of base instances with highest recognition-related similarity and localization-related similarity to the novel class 'Bird'. (b) 1-shot object detection results of randomly selected test samples by DeFRCN [26] and our approach on PASCAL VOC Novel Split 1. More examples can be found in the supplementary materials.

faceted commonalities, our object detector is able to perform recognition and localization more accurately, as shown in Figure 3(b).

5 Conclusion

In this paper, we propose the multi-faceted distillation for few-shot object detection. The key insight is to learn three types of commonalities between base and novel classes explicitly: recognition-related semantic commonalities, localization-related semantic commonalities and distribution commonalities. Then these commonalities are distilled during the fine-tuning stage based on the memory bank. Our method improves the state-of-the-art performance of few-shot object detection by a large margin.

Acknowledgements This work was supported in part by the NSFC fund (U2013210, 62006060, 62176077), in part by the Guangdong Basic and Applied Basic Research Foundation under Grant (2019Bl515120055, 2021A1515012528, 2022A1515010306), in part by the Shenzhen Key Technical Project under Grant 2020N046, in part by the Shenzhen Fundamental Research Fund under Grant (JCYJ20210324132210025), in part by the Shenzhen Stable Support Plan Fund for Universities (GXWD20201230155427003-20200824125730001, GXWD20201230155427003-20200824164357001), in part by CAAI-Huawei MindSpore Open Fund(CAAIXSJLJJ-2021-003B), in part by the Medical Biometrics Perception and Analysis Engineering Laboratory, Shenzhen, China, and in part by the Guangdong Provincial Key Laboratory of Novel Security Intelligence Technologies (2022B1212010005).

References

1. Cao, Y., Wang, J., Jin, Y., Wu, T., Chen, K., Liu, Z., Lin, D.: Few-shot object detection via association and discrimination. In: NeurIPS (2021)
2. Chattopadhay, A., Sarkar, A., Howlader, P., Balasubramanian, V.N.: Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In: WACV (2018)
3. Chen, H., Wang, Y., Wang, G., Qiao, Y.: Lstd: A low-shot transfer detector for object detection. In: AAAI (2018)
4. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International journal of computer vision **88**(2), 303–338 (2010)
5. Fan, Q., Zhuo, W., Tang, C.K., Tai, Y.W.: Few-shot object detection with attention-rpn and multi-relation detector. In: CVPR (2020)
6. Fan, Z., Ma, Y., Li, Z., Sun, J.: Generalized few-shot object detection without forgetting. In: CVPR (2021)
7. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: ICML (2017)
8. Girshick, R.: Fast r-cnn. In: ICCV (2015)
9. Hahn, S., Choi, H.: Self-knowledge distillation in natural language processing. arXiv preprint arXiv:1908.01851 (2019)
10. Han, G., He, Y., Huang, S., Ma, J., Chang, S.F.: Query adaptive few-shot object detection with heterogeneous graph convolutional networks. In: ICCV (2021)
11. Hariharan, B., Girshick, R.: Low-shot visual recognition by shrinking and hallucinating features. In: ICCV (2017)
12. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: CVPR (2020)
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
14. Hinton, G., Vinyals, O., Dean, J., et al.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 **2**(7) (2015)
15. Hu, H., Bai, S., Li, A., Cui, J., Wang, L.: Dense relation distillation with context-aware aggregation for few-shot object detection. In: CVPR (2021)
16. Kang, B., Liu, Z., Wang, X., Yu, F., Feng, J., Darrell, T.: Few-shot object detection via feature reweighting. In: ICCV (2019)
17. Karlinsky, L., Shtok, J., Harary, S., Schwartz, E., Aides, A., Feris, R., Giryes, R., Bronstein, A.M.: Repmet: Representative-based metric learning for classification and few-shot object detection. In: CVPR (2019)
18. Kim, K., Ji, B., Yoon, D., Hwang, S.: Self-knowledge distillation with progressive refinement of targets. In: ICCV (2021)
19. Komodakis, N., Zagoruyko, S.: Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In: ICLR (2017)
20. Lee, K., Maji, S., Ravichandran, A., Soatto, S.: Meta-learning with differentiable convex optimization. In: CVPR (2019)
21. Li, A., Li, Z.: Transformation invariant few-shot object detection. In: CVPR (2021)
22. Li, B., Yang, B., Liu, C., Liu, F., Ji, R., Ye, Q.: Beyond max-margin: Class margin equilibrium for few-shot object detection. In: CVPR (2021)
23. Li, Y., Zhu, H., Cheng, Y., Wang, W., Teo, C.S., Xiang, C., Vadakkepat, P., Lee, T.H.: Few-shot object detection via classification refinement and distractor retreatment. In: CVPR (2021)

24. Li, Z., Zhou, F., Chen, F., Li, H.: Meta-sgd: Learning to learn quickly for few-shot learning. arXiv preprint arXiv:1707.09835 (2017)
25. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014)
26. Qiao, L., Zhao, Y., Li, Z., Qiu, X., Wu, J., Zhang, C.: Defrcn: Decoupled faster r-cnn for few-shot object detection. In: ICCV (2021)
27. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: NeurIPS (2015)
28. Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: Fitnets: Hints for thin deep nets. arXiv preprint arXiv:1412.6550 (2014)
29. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision **115**(3), 211–252 (2015)
30. Salakhutdinov, R., Tenenbaum, J., Torralba, A.: One-shot learning with a hierarchical nonparametric bayesian model. In: ICML Workshop (2012)
31. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. In: NeurIPS (2017)
32. Sun, B., Li, B., Cai, S., Yuan, Y., Zhang, C.: Fsce: few-shot object detection via contrastive proposal encoding. In: CVPR (2021)
33. Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P.H., Hospedales, T.M.: Learning to compare: Relation network for few-shot learning. In: CVPR (2018)
34. Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al.: Matching networks for one shot learning. In: NeurIPS (2016)
35. Wang, X., Huang, T., Gonzalez, J., Darrell, T., Yu, F.: Frustratingly simple few-shot object detection. In: ICML (2020)
36. Wang, Y.X., Girshick, R., Hebert, M., Hariharan, B.: Low-shot learning from imaginary data. In: CVPR (2018)
37. Wang, Y.X., Ramanan, D., Hebert, M.: Meta-learning to detect rare objects. In: ICCV (2019)
38. Wu, A., Han, Y., Zhu, L., Yang, Y.: Universal-prototype enhancing for few-shot object detection. In: ICCV (2021)
39. Wu, A., Zhao, S., Deng, C., Liu, W.: Generalized and discriminative few-shot object detection via svd-dictionary enhancement. In: NeurIPS (2021)
40. Wu, J., Liu, S., Huang, D., Wang, Y.: Multi-scale positive sample refinement for few-shot object detection. In: ECCV (2020)
41. Wu, Z., Xiong, Y., Yu, S.X., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. In: CVPR (2018)
42. Xiao, Y., Marlet, R.: Few-shot object detection and viewpoint estimation for objects in the wild. In: ECCV (2020)
43. Xu, T.B., Liu, C.L.: Data-distortion guided self-distillation for deep neural networks. In: AAAI (2019)
44. 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: ICCV (2019)
45. Yang, S., Liu, L., Xu, M.: Free lunch for few-shot learning: Distribution calibration. In: ICLR (2020)
46. Yang, Y., Wei, F., Shi, M., Li, G.: Restoring negative information in few-shot object detection. In: NeurIPS (2020)
47. Yun, S., Park, J., Lee, K., Shin, J.: Regularizing class-wise predictions via self-knowledge distillation. In: CVPR (2020)
48. Zhang, C., Cai, Y., Lin, G., Shen, C.: Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In: CVPR (2020)

49. Zhang, L., Song, J., Gao, A., Chen, J., Bao, C., Ma, K.: Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In: ICCV (2019)
50. Zhang, L., Zhou, S., Guan, J., Zhang, J.: Accurate few-shot object detection with support-query mutual guidance and hybrid loss. In: CVPR (2021)
51. Zhang, W., Wang, Y.X.: Hallucination improves few-shot object detection. In: CVPR (2021)
52. Zhu, C., Chen, F., Ahmed, U., Shen, Z., Savvides, M.: Semantic relation reasoning for shot-stable few-shot object detection. In: CVPR (2021)