诚 信 声 明

我声明，所呈交的毕业论文是本人在老师指导下进行的研究工作及取得的研究成果。据我查证，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得其他教育机构的学位或证书而使用过的材料。我承诺，论文中的所有内容均真实、可信。

毕业论文作者签名：　　签名日期：　年　月　日

基于特征金字塔网络的少样本目标检测

[摘 要] 近二十年来，目标检测一直是计算机视觉和模式识别领域的研究热点。由于有足够的带标签的训练样本，基于深度学习的特征金字塔网络和 Faster-RCNN 等检测方法，在这一研究领域取得了很大的成功。然而，为了训练一个好的检测器而收集大量的目标框标签是非常耗时和昂贵的。由于只有少量带标签的数据用于训练，这些目标检测算法的性能可能不令人满意。虽然 MAML 算法对少样本图像的识别具有很好的性能，但它不是根据目标检测问题专门设计的，也不能直接用于检测。在本文中，我们提出了一种改进的基于 MAML 的少样本目标检测算法。该算法基于 Faster-RCNN 和 FPN 网络结构，对模型参数进行训练，使模型能够在少数几步的梯度下降和少量训练数据的情况下进行快速微调至好的结果。实验结果表明，该方法在 1-shot、5-shot 和 10-shot 的目标检测问题中均优于现有方法。

[关键词]少样本目标检测；未知模型元学习算法；特征金字塔网络；Faster-RCNN

Few-shot object detection based on feature pyramid network

Abstract： Object detection has been an active research topic in computer vision and pattern recognition since the last two decades. With sufficient annotated samples for training, deep learning based detection methods e.g., Feature Pyramid Networks and Faster-RCNN have achieved great success in this research area. Nevertheless, it is time-consuming and expensive to collect a large amount of bounding box annotation for training a good detector. With only a small amount of annotated data for training, the performance of these object detection algorithms may be not satisfactory. Though MAML algorithm has for few-shot image recognition to obtain better performance, it is not designed and cannot be employed directly for detection. In this paper, we propose an improved MAML based detection algorithm for few-shot object detection. In the proposed algorithm, based on the network structure of Faster-RCNN and FPN, the parameters of the model will be trained such that the model can quickly fine-tune in a small amount of gradient steps with a small amount of training data. Experimental results show that the proposed method outperform existing approaches for 1-shot, 5-shot and 10-shot object detection.

Keywords： few-shot object detection; Module-Agnostic Meta-Learning;

Feature Pyramid Networks; Faster-RCNN

目　　录

1 绪论

1.1 研究背景和意义

通过较少的样本能够快速识别不同的物体是人类的一种本能，人类能从几张有标签的图片迅速的认识图片中带标签的各种目标。我们期望机器也能够拥有这种能力，因此，图片分类、目标检测等任务应运而生。在以往的计算机视觉的研究当中，深度神经网络被广泛地作为“特征提取器”[1][2][3][4][5][6]，给予常规的深度神经网络大量的训练数据，一个性能较好的“特征提取器”就能被训练出来。但实际应用中，常常许多类别的样本只有少量，甚至只有几个或一个。如何训练出一个模型能够在只提供少量的样本的前提下，学习更多样本的信息，就是少样本学习问题所研究的。[7][8][9][10]

在少样本学习问题中，少样本目标检测任务受到了许多的关注，相对于少样本识别任务而言，少样本目标检测任务除了需要将图片中的每一个目标分类外，还需要精确地定位每个目标在图中的位置。这给少样本目标检测任务增大了难度。

1.2 研究现状

目标检测任务：RCNN 算法[11]是最早一批利用卷积神经网络进行目标检测的模型，该模型尽管使用了卷积神经网络，但仍然需要分两步进行目标检测，即先生成候选区域（可能为目标的区域）再检测是否为目标，以及该目标的具体类别；而后 SPP Net[12]又通过在输入全连接层前定义特殊的池化层打破了固定尺寸输入这一问题。后来一些更为准确的端到端的方法如 Mask

RCNN[2],Fast RCNN[13]和 Faster RCNN[14]，提倡使用单一尺度计算特征，因为这样能够在准确性和速度上有一个较好的平衡。但这也导致了小对象检测的准确率低。近些年，特征金字塔结构的提出能够跨多个尺度识别对象，但由于该结构需要大量的计算机内存进行模型训练，导致金字塔结构在很长一段时间仅仅应用在一些轻量级的任务中。而 FPN 算法的提出不仅结合了特征金字塔结构多个尺度识别对象，还能在时间与内存的消耗上优于其他使用特征金字塔结构的算法，使特征金字塔结构能够应用于较为大型的目标检测任务中。

少样本识别任务：最近的少样本识别任务中，根据 Oriol Vinyals 在 NIPS 2018 元学习研讨会上的总结，可以将元学习算法可以分为 3 类：基于度量、基于模型以及基于优化。基于度量的元学习算法的思想来源主要是最近邻算法，通过度量学习的方法生成输入图片的嵌入向量，并设计相应的内核函数，以此来学习两个样本之间的相似性。卷积暹罗神经网络[15]、Matching Network[22]等都是这种方式的代表。而基于模型的元学习算法[26]则是训练一个专门为快速学习设计的模型，该模型的特点为只需要几个训练步骤就能快速更新参数，以这种方式即能在少数的样本训练下得到一个在测试集上有较优结果的模型。本文借鉴的元学习方法是基于优化的元学习算法，Ravi & Larochelle[16]将梯度学习算法优化显示建模，并命名为“元学习者”，由于 LSTM[15][17]小区态更新与反向传播中的基于梯度更新有相似之处，因此将“元学习者”建模为 LSTM。基于此，Ravi & Larochelle[16]将 LSTM 算法应

用在元学习中，训练一个基于 LSTM 的元学习器来学习网络的更新规则。把 LSTM 中的单元状态看作元学习器的参数。目的是使元学习器通过训练，而确定学习率和遗忘门的参数最优值。学习到的初始值能让元学习者确定最佳初始权重，帮助优化快速进行。后来 Finn[9]等人提出了一种简单并且适用于几乎所有通过梯度下降学习的模型的未知模型元学习（Model-Agnostic Meta-Learning）算法，这一方法中每一个元任务独享一个网络参数，在梯度下降更新完全部任务的网络参数后，通过对这些网络参数进行梯度下降更新整个网络，这样有助于训练出来的网络能够快速适应每一个任务。

最近的研究在使用深度神经网络进行少样本学习中取得了重大的进展，并在分类，回归等问题上得到了证明[9]，然而，关于目标检测问题的研究仍然是少数。目标检测问题存在图像背景和需要定位的目标，使得通过几个样本，识别一个类别实例变得复杂，传统的少样本学习方法难以解决目标检测中的定位目标的任务。

1.3 本文的主要工作和创新点

在本文中，我们提出一种适用于目标检测的少样本学习算法，并证明该算法不仅能够在目标分类上有效，同时，在目标定位上的效果也有提升。我们的算法借鉴 MAML 算法[9]，能够很好的应用于训练深度神经网络模型，在元学习中，不再需要像以往的元学习更新函数，只需要在一个新任务上通过一个或几个梯度下降更新网络参数，这一步骤可以认为是优化模型过程，因此该模型的底层表示适应各种不同的任务，只需要在遇到新的少样本目标检测任务中对参数进行微调就能得到良好的结果。

该算法的网络结构使用了 FPN[18]中的特征金字塔结构用来解决 Faster-RCNN 算法在单一尺度上进行检测的问题(实现了多尺度目标检测),Faster-RCNN 算法依赖 RPN 算法生成可能为目标的区域，并使用一个分类器将这些可能为目标的区域划分为目标和背景两类,在对目标进行分类。在我们的方法中,先将少样本检测任务分为目标定位任务和目标分类任务两个任务。对于目标区域生成任务，我们将真实区域与预测区域的转化关系以参数表示，在训练是设置多个少样本训练任务对该参数进行训练，使该转化关系在元训练阶段能够有更好的适应性，只需要使用少量的新样本对模型进行微调就能使得该模型在新样本目标区域生成有良好的结果。目标分类任务,我们则是将整个网络的参数通过多个少样本训练任务进行训练，使得整个网络对于新类识别有较好的适应性。因此,该方法能同时提高目标区域生成任务和目标分类任务的准确率。

我们在少样本目标检测实验中，通过使用两个不同的数据集来验证我们算法的有效性，并通过与其他少样本目标检测方法比较 mAP 展示我们算法在准确率和回归率等方面的优越性。

本文的主要贡献是提出了一种简单且高效的少样本目标检测方法，将原本用于分类的少样本学习算法 MAML 的少量的梯度更新带来的新任务的快速学习的思想应用于目标检测任务中，这样无论是在目标定位任务或是目标分类任务在样本少的情形下仍然得到一个在新任务有高适应性的模型。

1.4 本文的研究框架

本文接下来的章节安排如下:第 2 章简要的回顾了与本文相关的工作、

这些工作的优缺点以及一些在本文中提及的基础概念的介绍。第 3 章提出基于特征金字塔网络的少样本目标检测算法，在提出该算法前还介绍了我们训练和测试中的任务设置。第 4 章首先介绍我们实验中实验的数据集、将要作为对照实验的有代表性的基线以及我们的评价指标，通过将我们的实验在相同数据集上的实验结果与我们挑选的几个具有代表性的基线的实验结果进行对比，验证了我们实验的优越性，而后，对我们的算法进行进一步的分析，验证我们算法在全部类别中（包括基类和新类）的有效性以及将算法分为两个阶段实现的必要性。最后一个章节是对整个文章进行简要的总结，以及对未来的研究工作的展望。

2 基础概念与相关研究

2.1 元学习算法

我们借鉴了模型未可知元学习算法[9]的主要思想，将原本应用于目标分类的算法经过改进应用于目标检测中。以下是在目标分类任务中模型未可知元学习算法[9]（MAML）的介绍。

MAML 算法是通过优化网络初始化参数的更新机制来使神经网络有更强的学习能力。首先是将图片按类别分类后，随机在各个类别上不断的取若干张图片组成多个任务的训练集和测试集，为了区分，我们称训练集为支持集，称测试集为目标集。其中测试集的类别标签用于计算经过支持集训练后模型在测试集的表现（训练后的模型的表现以损失函数的形式表示），在元训练阶段，MAML 算法通过这些支持集的输入对参数进行第一轮的更新，这一轮的更新并没有作用于原模型，而是用于计算第二轮的新参数，即利用第一轮更新后的参数，结合目标集来计算第二轮的梯度，而这一轮的梯度直接作用于原模型的参数更新。而在微调（fine-tune）阶段，则是在新任务中进行第一轮的参数更新，而后将更新结果直接作用于原模型，此时更新后的结果被证明能够在新任务上有好的表现（在目标分类中模型的评价指标常常是准确率）。

但该方法适用于分类、回归或者是强化学习，而在目标检测任务中并不能直接改良目标区域生成任务，并且在检测的目标分类任务中以往的损失函数不再适用。

2.2 特征金字塔网络结构

特征金字塔网络（FPN[18]）是为了解决以往特征提取器在小尺度或中尺度中检测结果不好的问题。在提出该网络结果以前，多尺度的目标检测往往是通过将图片缩小或者扩大作为输入来反映不同尺度的特征组合。但这会对计算机内存造成巨大的负担，因此适用的领域很小。

而 FPN 使用卷积神经网络（CNN）中每一层的信息来生成所需的特征组合，原文中以图 1 表示其基础架构。每一层的网络结构都表示不同尺度的特征信息。该基础架构主要分为三个部分：自下而上的不同尺度特征信息生成（bottom-up 网络）[19]，自上而下的特征信息补充加强以及自下而上的卷积神经网络与自上而下得到的各维度特征的关联表达。

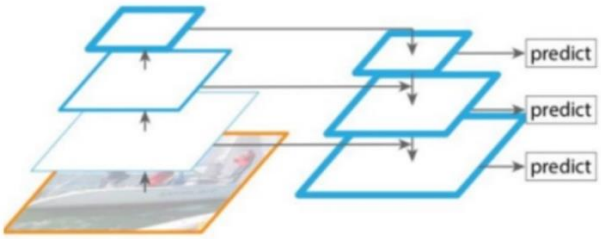

图 1：特征金字塔网络结构[18]

首先是自下而上的路径， 这一个过程即是 CNN 特征不断的进行卷积浓缩表达特征，在原文中，作者使用了残差神经网络作为这一过程的网络。在每一层的特征表达中，特征图在经过卷积核扩张卷积[23][24]后特征层的输出大小会和原本的大小一样，如图 2 所示，这是一个特征层不断浓缩的过程，在原文中，作者使用每一层的最深层作为下一层的输入，因为在这一层中具有最强的特征。这一自底向上的过程很早就已经被认识到了，即较底的层反映较浅层次的图片特征信息如边缘等，较高的层则反映较深层次的图片特征信息如类别等。

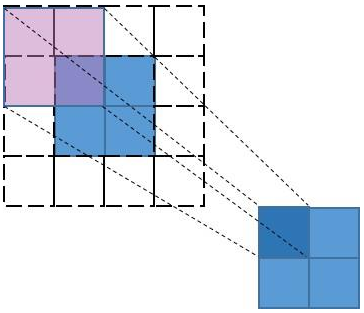

图 2：通过扩张卷积[23][24]使得输出的特征大小不变，该图为 2 * 2 的特征图经过 2 * 2 的卷积核扩张卷积得到 2 * 2 的特征图

自上而下的路径与自下而上的卷积神经网络与自上而下得到的各维度特征的关联表达，这两部分是对特征进行增强，原文[18]是通过以下的方法实现的，由于处于金字塔越高层所得到的特征图是更加抽象且包含更多信息的，因此作者在处理时通过将该层结构的上层进行上采样操作，同时前一层的特征需要经过 1 * 1 的卷积核卷积后与上采样后的特征图进行横向连接，使高层特征得到加强。最后，原文中作者为了消除上采样的混叠现象（由于采样点减少出现的欠拟合问题），使用 3 * 3 的卷积核去处理已经融合的特征图，得到的特征图再应用到以往的各种目标检测算法中即可。

这种特征金字塔网络结构克服了以往目标检测在多尺度特征中表现差的问题，同时克服了以往特征金字塔结构提取特征带来的内存需求过大的问题。但在少样本目标检测上仍然起不了良好的效果。

2.3 区域候选框生成网络

区域候选框生成网络（RPN）是 Faster RCNN[14]提出的用来提取候选框的一种网络结构。我们截取了在 Faster RCNN[14]原文中表示 RPN 的结构的图如下

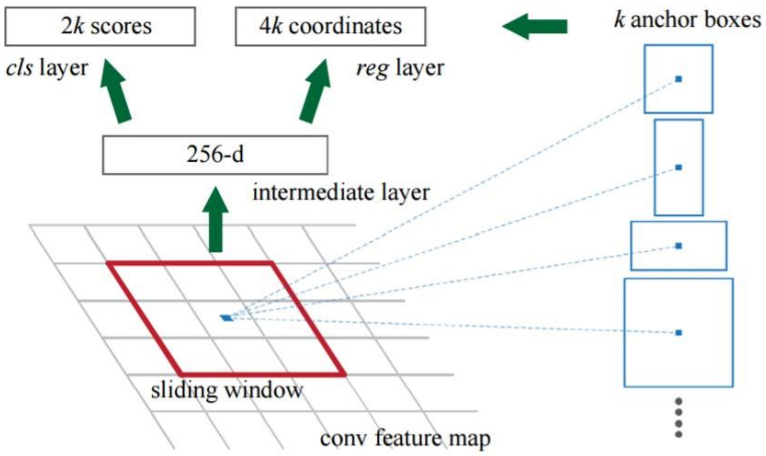

图 3：RPN 网络结构[14]

图中展示了 RPN 的整个过程，RPN 的输入是一个特征图（可以使用通过 FPN 处理后得到的特征图），我们假设这个输入的特征图有 N * M 个向量，每个向量是 D 维，在文中[14]（图 3），D 为 256 维。这个特征图经过一个 3 * 3 的滑动窗口（sliding window）的处理，就能得到 D 维的特征，然后通过两个全连接层的处理就能够得到 2k 个分类的得分和 4k 个回归的坐标，其中 k 是指由特征图锚点选出来的 k 个目标框，RPN 的工作就是判断这些框是不是前景，以及与真实的框（在 Faster RCNN 原文中也称为 Ground True）的偏移，因此 2k 中的 2 就代表这是一个二分类的任务（前景和背景两类），以及 4k 中的 4 是指每个框的四个坐标。需要注意的是 k 个框的大小和长宽比是提前给定的，在原文中共有 9 种组合，也就是 k = 9。

3 任务和方法

为了更好的进行少样本目标检测，我们将数据集按类别中带标签的多寡将类别分为基类和新类，其中，基类是指这些类别有大量的带标签的样本，而只有少量的带标签的样本的类别则为新类。我们的目标是通过训练基类获得一个在同时有基类和新类时有良好表现的少样本目标检测模型。

在以往的目标检测任务中，常常采用这样的设置，通过拥有大量的带标签的数据训练一个目标检测器，但对于实际任务而言，常常一个类别中的样本数有限，一般而言通过大量带标签的数据训练出来目标检测器在只有少量带标签的类别上效果很差。因此解决这种少样本目标检测问题是实际任务中迫切需要的。

3.1 少样本目标检测任务设置

首先，我们将数据集按照类别的分布划分为一个个任务，如图 4 所示。在元学习领域的论文中经常使用 k 和 N，k 代表少样本学习器在每个类别上学习的机会，N 代表少样本学习器需要学习的类别数量。对于目标检测任务而言，一个图片中一个目标作为一个样本，因此，如果一个元学习任务设置为“两步五次”，即 k=5，N=2，那么这一个元任务的支持集（为了区别整个目标检测任务中的训练集而将每一个元任务中的训练集命名为支持集）共有两个类别，每个类别五个目标，共有 10 个目标，在本文中也会出现 k-shot 和 N-way 的表示方式，这种表示方式与“k 步 N 次”的意义相同。

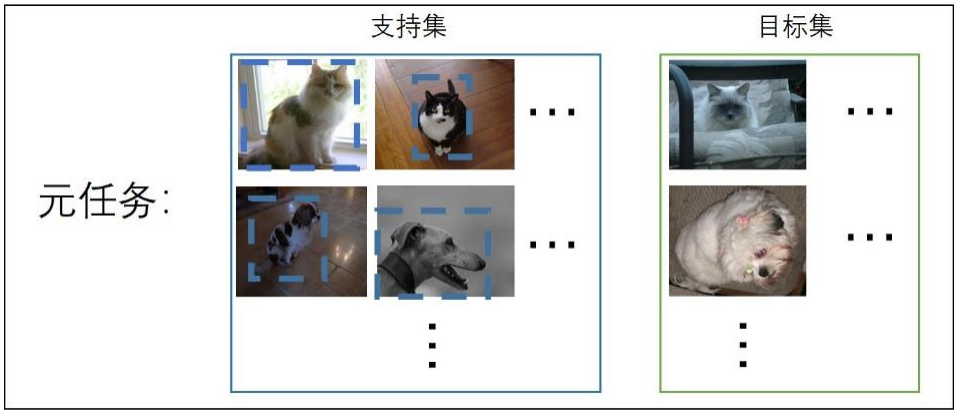

图 4：训练以及测试都以元任务的方式进行。这种训练模式最早由 Oriol Vinyals[22]，为了区别整个模型的测试集和训练集，我们将元任务中的训练集称为支持集，元任务中的测试集称为目标集

3.2 学习器训练

区别以往直接以带标签与目标区域标签的图片直接作为训练集，在我们的方法中，我们将元任务的集合作为训练集。在元训练任务中，我们随机从基类（大量的带有注释的数据）集合中挑选 N 个类别，每个类别挑选共带有 K 个目标（一个图片若有两个目标，则该图片当做两个目标处理）的样本作为支持集，并且从基类中随机挑出 N 个类别中一定数量的样本作为目标集。

首先，我们初始整个网络的参数θ和ω，其中θ是目标区域生成任务部分网络的参数，ω是目标分类任务部分网络的参数，由于目标分类任务的完成需要整个网络结构的参与，因此参数ω也可以作为整个网络结构的参数（即参数θ为参数ω的一部分）。每个任务都通过该网络训练各自的网络参数，这一过程是并行的，任务之间的网络参数并不互用，而每一个任务又将分为目标区域生成任务以及目标分类任务，需要注意的是尽管我们的算法中将任务分为了目标区域生成任务以及目标分类任务，但是这两个任务都在同一个网络结构中完成。

我们将数据集以随机的方式分为若干个元任务，设第i个元任务为T_i，T_i中的支持集中的图片处理后送入预训练网络中，即 bottom-up 网络[19]，然后经过自上而下网络，把高层特征图进行上采样操作，同时前一层（即金字塔结构中自下而上部分网络）的特征需要调节通道数（以 1 * 1 卷积核对不同通道的特征图线性组合的方式）后与上采样后的特征图进行横向连接，进一步增强较高层的特征表达。经过三个部分组成的金字塔结构[18]后，能够提取不同尺度的图像特征，将不同尺度的图像特征分别进行 RPN 操作[14]，即使用 3*3 的卷积核处理通过前面不断重复迭代进行的上采样样以及与横向连接像素点相加这一过程得到的精细的特征图，以得到一个通道数目为 256 的特征图，尺寸大小与卷积前的特征图大小相同，后通过全连接得到每一个 anchor 的前景分数、背景分数以及坐标，经过回归，得到候选目标区域。

我们将 FPN[18]中一层一层提取的不同尺度的特征的金字塔结构作为直接影响目标区域生成任务的因素，也就是在目标区域生成任务中，我们通过梯度下降法只需要修改每一个元任务的网络中的金字塔结构部分网络的参数，我们设这部分网络参数为ω。梯度更新中目标区域生成任务的损失函数为：

$$\mathcal{L}_{T_i}(f_{RPN}) = \frac{1}{N}\sum_{j}(L_{cls}(p_j, p_j') + \lambda p_j' L_{reg}(t_j, t_j'))$$

其中，L_{cls}是分类损失函数，L_{reg}是回归损失函数，$\mathrm{p_i}$和$\mathrm{p_i'}$的值为 0 或 1，0 表示该区域为背景，1 表示该区域为前景。$\mathrm{P_i}$为预测值，$\mathrm{p_i'}$为真实值。而$\mathrm{t_i}$为目标区域候选框与目标区域的转化关系，其具体定义参考[14]，$\mathrm{t_i'}$为目标区域

候选框回归的目标，其定义同样参考[14]。

由于在 RPN 中，分类任务具体为分类该区域为前景或背景，即$\mathrm{p_i}$和$\mathrm{p_i'}$的值为 0 或 1，因此L_{cls}为二分类问题的损失函数，我们这里设置为交叉熵损失函数。而L_{reg}则是目标检测任务中常用的 smooth L1 loss 函数:

$$\mathrm{smooth_{L_1}}(x)=\begin{cases}0.5x^2, if\ \ |x|>1\\ |x|-0.5, otherwise\end{cases}$$

这样一来，在不同尺度下，我们都能得到若干个候选目标区域；对不同尺度的候选目标区域进行目标区域候选框池化，即 range of interest（ROI）pooling[18]操作，具体为将每个候选区域均匀分为 M×N 块，对每块进行最大池化层处理，能够将特征图上尺度不同的候选区域变为大小一致的数据，最后进入全连接层处理后就能得到回归和分类结果。

我们在目标分类任务中，通过梯度下降法修改每一个元任务的网络所有参数，目标分类任务的损失函数为：

$$\mathcal{L}_{T_i}(f_{Faster-RCNN})=\frac{1}{N}\sum_j(L_{cls}(p_j^*,p_j^{*\prime})+\lambda p_j' L_{reg}(t_j,t_j'))$$

其中p_j^*为目标预测类别，$p_j^{*\prime}$为目标类别的真实值。

与目标区域生成任务中的 RPN[14]的构成相同，都是由一个分类损失函数，和一个回归损失函数的组成，回归损失函数是相同的，因为这里的回归任务是对目标区域候选框的位置再次进行微调，而分类损失函数则稍有不同，在目标区域生成任务中，同样使用交叉熵损失函数作为损失函数，但分类任务是分类前景与背景。而在目标分类任务中，分类任务是区分不同目标的类别，因此两个分类损失函数的输入不同。

算法

需要：$\alpha,\beta,\gamma,\delta$四个超参数作为学习率
需要：n: 候选目标区域的个数
1: 随机初始化目标分类任务网络参数 ω
2: 随机初始化目标区域生成任务参数θ
3: **while** not done **do**
3: 根据任务要求随机生成任务 T_i
4: **for** all T_i **do**
5: 通过K个样本计算目标区域生成任务的梯度 $\nabla_\theta \mathcal{L}_{T_i}(f_{RPN})$
6: 通过K个样本计算目标分类任务的梯度$\nabla_\omega \mathcal{L}_{T_i}(f_{Faster-RCNN})$
7: 用梯度下降法计算RPN算法以及Fast-Rcnn算法的自适应参数
$\theta_i' = \theta_i - \alpha \cdot \nabla_\theta \mathcal{L}_{T_i}(f_{RPN})$
$\omega_i' = \omega_i - \beta \cdot \nabla_\omega \mathcal{L}_{T_i}(f_{Faster-RCNN})$
8: **end for**
9: 更新$\theta \leftarrow \theta - \gamma \cdot \nabla_\theta \sum_{T_i} \mathcal{L}_{T_i}(f_{RPN})$
$\omega \leftarrow \omega - \delta \cdot \nabla_\omega \sum_{T_i} \mathcal{L}_{T_i}(f_{Faster-RCNN})$
10: **end while**

除去网络结构（FPN + Faster RCNN）我们的整体算法参考**算法**，我们将算法结合网络结构后整理如图 5 所示，以上是通过多个元任务进行第一轮的网络参数更新，但这一轮更新的网络参数是隶属于不同元任务的，藉由不同元任务中的支持集训练网络参数，若我们将第 i 个元任务T_i的区域生成任务中的网络参数设为θ_i，第 i 个元任务T_i的目标区域分类任务中的网络参数设为ω_i，更新后的区域生成任务中的网络参数设为θ_i'，更新后的目标区域分类任务中的网络参数设为ω_i'，则有：

$$\theta_i' = \theta_i - \alpha \cdot \nabla_\theta \mathcal{L}_{T_i}(f_{RPN})$$

$$\omega_i' = \omega_i' - \beta \cdot \nabla_\omega \mathcal{L}_{T_i}(f_{Faster-RCNN})$$

其中，α，β是学习率。

为了使模型有更好的泛化能力，我们的模型需要进行第二轮的梯度下降更新参数，每一个任务更新后的网络参数作为第二轮梯度下降中的损失函数的参数，用不同元任务的目标集进行损失函数梯度的计算。将每个元任

务梯度下降法计算出来的值求和作为整个网络参数（目标区域生成任务中网络参数为θ，目标分类任务中网络参数则为ω）更新的梯度，即：

$$\theta \leftarrow \theta - \gamma \cdot \nabla_{\theta} \sum_{\mathrm{T_i}} \mathcal{L}_{\mathrm{T_i}}(f_{RPN})$$

$$\omega \leftarrow \omega - \delta \cdot \nabla_{\omega} \sum_{T_i} \mathcal{L}_{\mathrm{T_i}}(f_{Faster-RCNN})$$

其中，γ，δ是学习率。

这么做的目的是训练出网络参数θ，ω能够在新任务（含有新类的元任务）进行微调下快速更新至适合新任务的参数。

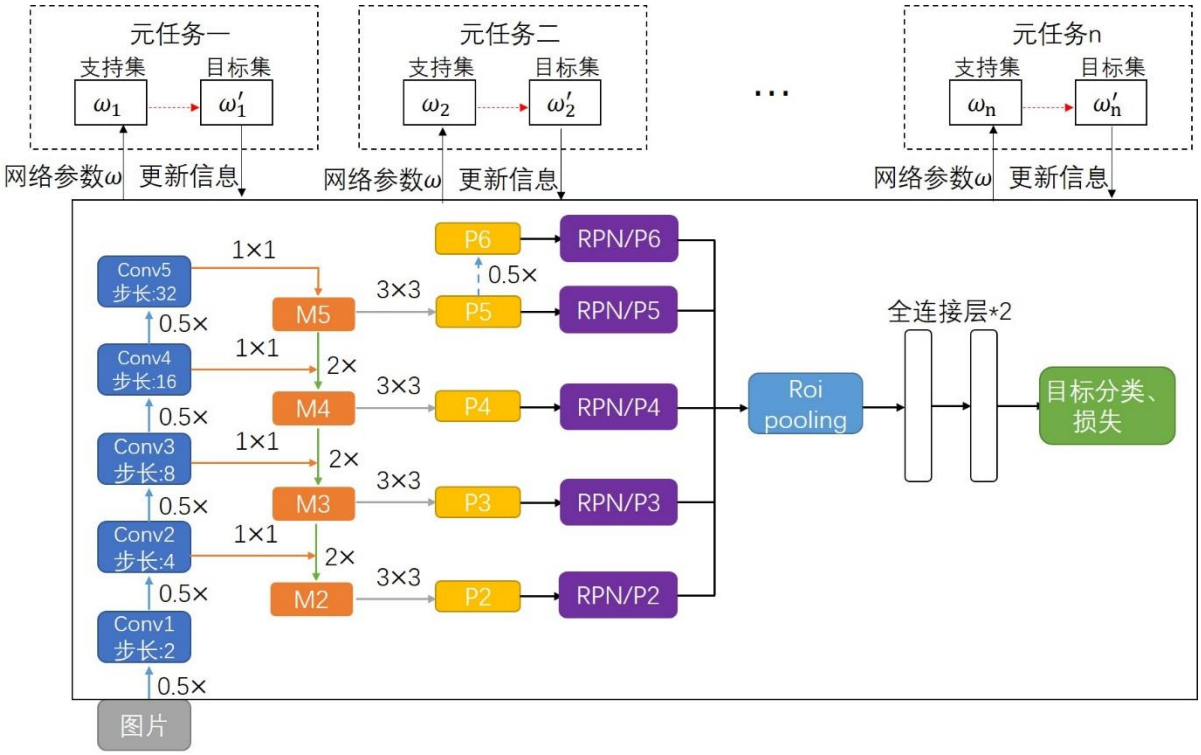

图 5：图中 0.5×表示的是降采样操作，1×1 则是将特征层经过 1×1 卷积核进行卷积产生新的粗略的特征图，2×则是将高层特征做两倍上采样操作（本文中使用的是最近邻上采样法），将上采样与横向 1×1 卷积后的特征图进行特征结合，才能得到图中的橙色框代表的特征图。而 3×3 则是表示将特征图进行 3×3 卷积。

也因此整个模型分两个阶段训练，两个阶段的训练所使用的网络模型相同，第一个阶段的训练是在大量的基类样本中进行训练，第二个阶段的训练则是在少量的新类样本中微调模型参数，使微调后的模型能够在新类有好

的目标检测结果，在第二阶段的训练中，与第一阶段不同的是，第一阶段每一个元任务中的支持集训练并不直接影响原模型网络的参数，而是在目标集测试中才通过各个元任务更新后的信息作为原模型网络更新的方向；而第二阶段中的每一个元任务中的支持集训练后是直接影响原模型网络的参数的，因为只有这样，才能够通过少量样本的微调使模型在新类中也能够有好的表现。

4 实验：

在本文中，为了评估我们的模型效果，我们将模型的结果与各种基线进行对比，以此表明我们的模型如何更准确地检测新对象（只具有少样本的类别），我们使用 YOLOv2[23]作为基探测器，并且以目前的热门研究[7]中的实验结果进行对比。

4.1 实验设置

4.1.1 数据集

本实验中用于少样本目标检测任务的数据集为 VOC2007 以及 VOC2012 两个数据集，我们复现了以往研究中的多个实验，并使用 VOC2007 以及 VOC2012 两个数据集的训练集进行训练，VOC2007 的测试集作为实验的测试集进行模型测试。在实验中，我们需要人为的设置几个类别作为新类，而 PASCAL VOC 数据集的样本中一共具有 20 个类别，为了与将 Feature Reweight 应用于目标检测这一先进方法[7]中的实验有较好的对照结果，我们随机选择 5 个类作为实验中的新类，其余 15 个类作为基类。我们评估了 3 种不同的类划分的方式。在模型微调部分，我们使用少量的数据集进行训练，保证新类的训练中每一个新类中的所有图中只有 K 个该类目标，在我们的实验中，K 被设置为 1、5、10。

需要注意到，测试图像中有着含有新类同时包含着基类的图片或者不含新类的图片；因此，当新类样本数量较少时，新类的目标检测任务变得更加困难。

4.1.2 基线

我们将我们的模型与 Feature Reweight[7]等 3 个有竞争力的基线进行比较，由于现阶段少样本目标检测相关研究尚少，我们希望观察当前在目标检测中优异的模型在少样本目标检测任务中是否有较好的结果。于是我们选择了一个当前在目标检测领域取得优异结果的基线 YOLOv2[23]并将其应用于少样本目标检测任务中，YOLOv2 是在 YOLOv1[24]的基础上结合 Faster RCNN[14]进行改进，具体是将大量基类的图片和少量新类的图片一起训练检测器，通过这种方法，可以从基类中学习到用于检测新类的特征，而这条基线被称为 YOLO-joint[7]。此外，为了更好地说明我们实验分为两个阶段的合理性，我们将两个阶段的训练合并为一个阶段，将大量的基类和少量的新类作为训练集进行训练，我们称这条基线为 FPN-joint。最后，我们还找了一个在少样本目标检测领域取得优异结果的基线 Feature Reweight[7]与我们在少样本目标检测任务中取得的实验结果进行对比，该实验先是利用有大量标签的基类训练一个特征调整模块，通过这个模块可以用这些基类的底层特征对待检测图片的特征进行调整。这些底层特征一般是反映所有类通性的特征，然后，再将网络经过第二阶段的训练微调到新类的检测中去。

为了更好地进行对照实验，我们令基线与我们的模型的训练总迭代数相同。

4.1.3 模型评价指标

由于我们使用 PASCAL VOC 数据集，对于该数据集，PASCAL Visual

Objects Classes(VOC)竞赛设置了评价模型好坏的指标 mean Average Precision（mAP）[25]。该指标能够同时反映一般机器学习模型的查准率 Precision 和查全率 Recall，并且能够针对目标检测的目标区域生成任务完成情况作出评价，是目标检测中较好的评价指标。因此我们使用该指标作为模型评价指标。

为了计算 mAP，我们需要如其他机器学习问题一样，需要知道我们检测结果的 True Positives（TP）、False Positives（FP）、True Negatives（TN）和 False Negatives（FN）用以计算 Precision 和 Recall。

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

我们首先通过标签划分一个图片中的正样本和负样本，对于正样本我们需要区分 True Positive 和 False Positive，在目标检测任务中，这主要通过 Intersection over Union （IoU）来得到。其中：

$$\text{IoU} = \frac{\text{真实框与预测框交集}}{\text{真实框与预测框并集}}$$

以图示的方式 IoU 表示为图 6：

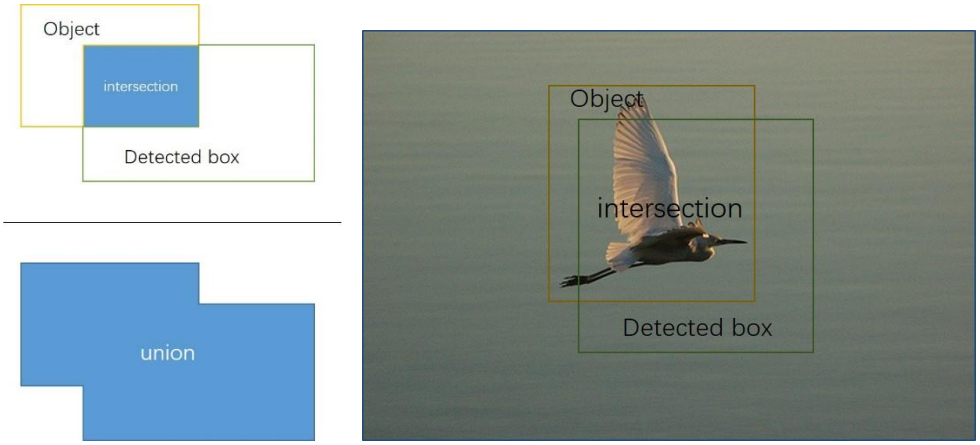

图 6：图中 Object 所在的黄色边框是真实框，也就是目标解，而 Detected box 则是模型预测出来的预测框，intersection 是两个框的交集，union 则是两个框的并集。两个框的交集与两个框的并集的比值即为 IoU。

计算出 IoU，我们可以确定一个检测结果为 Positive 是正确的（TP）还是错误的(FP)，最后计算出模型漏检的目标(FN)，我们就能够得到 Precision 和 Recall，将模型预测结果按照各个预测值置信度降序排列，改变置信度阈值，得到多个不同的 Precision 和 Recall 并画出 PR 曲线（Recall 作为横轴，Precision 作为纵轴），每个类别的 PR 曲线与横坐标轴所形成的面积的近似即为 Average Precision（AP），而我们所求的 mAP 则是多个类别的 AP 的平均值。

4.2 与基线实验结果的对照

在 PASCAL VOC 数据集上的对照实验。我们的对照实验结果在表 1 中显示，首先，我们清楚地发现我们的模型明显优于三条基线，特别是在标签数极少的情况下，即当只有 1 个标签进行有监督学习时，我们的模型比基线的结果都要优异，这符合我们对于少样本目标检测模型的预期。不仅如此，我们的模型在标签数增加到 5 或者 10 的情况下都普遍优异于基线。我们认为在未分成两个阶段前，由于在训练的过程中大量的基类标签会将少量的新类

带来的偏向削弱，导致出现许多新类漏检或者是错检的情况出现。从表 1 中也可以看出，由于我们设置了 FPN-joint 这一条将两个阶段的训练合并为一个阶段的基线，训练的结果显示我们的模型分为两个阶段的训练后在 1-shot、5-shot 和 10-shot 下的 mAP 均有较大的提升，因此我们的假设成立，这也意味着模型分为两个阶段训练是必要的。此外，对于当下在少样本目标检测领域有竞争力的研究 Feature-Reweight[7]的表现而言，我们的模型在总体上占优，仅仅是在第二个随机挑选的新类集中 5-shot 的实验结果略逊与该基线的结果，其余实验结果我们的模型均有较为明显的提升。

	Novel Set 1			Novel Set 2			Novel Set 3		
Method / Shot	1	5	10	1	5	10	1	5	10
Yolo-joint[23]	0.0	0.0	0.0	0	3.6	3.6	0	0.3	0.7
FPN-joint	5.4	13.9	24.2	4.0	12.6	25.6	3.7	9.8	9.8
Feature-Reweight[7]	10.5	35.4	42.9	12.4	**32.6**	40.9	15.5	38.5	48.3
our	**19.3**	**39.1**	**46.3**	**13.7**	31.9	**42.5**	**16.3**	**40.8**	**50.8**

表 1：少样本目标检测在 PASCAL VOC 数据集上的表现（mAP）。我们在三个不同的新类设置上比较少样本检测器的好坏，结果表明，我们的检测器基本优于基线的结果。

4.3 算法分析

我们将训练模型改为直接使用该网络结构进行少样本目标的训练学习，这种方式能够直接地看出我们算法训练自适应参数的必要性，若是不使用这种元学习模型，直接将少量的新类样本以及大量的基类样本送入 FPN 与 Faster-RCNN 结合的网络结构中进行训练以及进行第一阶段训练后，第二阶

段不采用元学习的方式进行 fine-tune。得到的实验结果与我们的模型对比如表 2 所示。

		Novel Set					Base Set														
Shot	Method	aero	bird	bus	dog	person	bike	boat	bottle	car	cat	chair	cow	table	horse	mbike	plant	sheep	sofa	train	tv
5	FPN[18]	17.1	3.8	10.7	5.3	0.4	**79.9**	**65.1**	**48.5**	**80.6**	**79.7**	**59.4**	**73.8**	**73.0**	**80.4**	**78.9**	**54.6**	**70.3**	67.9	**75.9**	**70.9**
	Our	**54.2**	**32.5**	**54.3**	**32.1**	**22.6**	70.0	55.1	37.3	73.1	65	48.3	68.4	62.9	72.0	74.0	45.6	59.1	**71.9**	70.1	69.5
10	FPN[18]	37.9	16.1	27.4	11.4	8.6	**81.3**	**61.3**	**53.9**	**78.1**	**80.7**	48.6	65.6	**74.6**	**76.0**	**77.4**	**58.5**	**67.3**	**74.8**	**77.8**	**72.5**
	Our	**58.6**	**44.1**	**59.3**	**39.7**	**29.8**	75.1	57.4	39.8	75.3	71.1	**52.5**	**67.6**	59.7	73.5	73.2	48.7	60.8	67.1	74.9	69.3

表 2：使用FPN[18]进行特征提取与我们的目标检测器在各个类上的表现（mAP），在新类上，我们的模型有绝对的优势，总平均表现同样具有优势。

首先，由表 2 可知，只使用 FPN 与 Faster-RCNN 结合的网络结构处理新类的目标检测的效果差，基本不能够正确地识别目标。而我们的模型在 5-shot 和 10-shot 两种情况下在新类的效果都要远远的优于 FPN 处理的结果。即有无对模型的自适应参数进行训练对于模型有着较大的影响。同样的，我们发现在基类上，我们的模型的表现会有一定程度的变差，其原因我们会在下文进行进一步的分析研究。但尽管在基类上的表现会有一定程度的降低，但是在 20 个类的总的 mAP 上，我们的模型表现分别为 56.9mAP、59.9mAP，而 FPN 与 Faster-RCNN 网络结构下测试后的结果分别为 54.8mAP、57.5mAP，因此，我们的模型总体上仍然有提升。

同时，除了分析在新类上能够有比基线更好的目标检测结果，还需要保证在基类在经过新类进行微调训练后的目标检测结果不至于变得太差。理想情况下，在新类微调训练前后，在基类上的训练结果应该一致。我们比较

了第二阶段我们的模型在新类训练后更新的模型与基线中采用的算法在基类的训练结果在 1-shot、5-shot 和 10-shot 实验中的平均值。结果如表 3 所示，表明我们的模型虽然是少样本目标检测模型，但在基类上仍然有良好的表示结果。这为解决少样本目标检测问题奠定了基础。

Step2	Base Set 1	Base Set 2	Base Set 3
Feature-Reweight[7]	60.7	**62.0**	59.3
Ours	**63.5**	61.7	**59.7**

表 3：我们使用 mAP 来评判模型的表现，我们发现在第二阶段我们的结果与基线的结果相差不多，甚至优于基线的结果。

我们猜测在前文中提及到的我们的模型在基类上结果会削弱的现象是由第二阶段对于模型网络参数在新类上再调整有关。因此我们还比较了我们的模型第一阶段后的基类的训练模型与 FPN[18]模型在相同测试集上测试的结果（如表 4）。从结果中可知，我们的算法在基类上训练的结果与 FPN[18]的网络结构应用在 Faster-RCNN[14]中的结果相差不大，也就是说我们的算法对于 FPN 网络结构的优势发挥并没有太大的影响。

	Base Set 1	Base Set 2	Base Set 3
FPN[18]	**70.9**	71.3	**70.6**
Ours（step1）	70.8	**73.2**	70.2

表 4：目标检测器在基类的检测表现（mAP），我们比较了我们的模型的第一阶段与 FPN[18]的表现。

结论

在我们的实验中，我们提出了一种能够适用于目标检测的改进后的 MAML 算法并结合特征金字塔网络结构进行目标检测。为了解决以往少样本目标检测任务中仅仅提升目标分类的而对于目标区域生成无直接提升的问题，我们的算法将两个任务放在同一个网络中，但是分开训练。以这种形式达到对目标分类和目标区域两方面都能够有所提升。此外，我们通过将训练划分为两个阶段，分别为在基类上进行的第一阶段训练以及在新类上进行的第二阶段的训练。实验表明，这种分段训练的方式有助于少样本目标检测任务。在文中，我们的实验结果在 PASCAL VOC 数据集上与 3 个先进的实验结果对比证明了我们的方法的优越性和有效性。本文并没有对于前人所用的无论是“特征提取器”或是“目标分类器”的网络结构进行改进，因为我们未来的工作是针对整个算法的网络结构能够有进一步的改进，尤其是目标区域生成部分的网络，希望网络不仅能够适应不同尺度，同时在不同尺度的目标上有进一步的辅助机制提高目标区域生成准确率。

致谢

时间一晃，就到了毕业的季节了，大学四年也终于要告一段落了。这四年，有遗憾、有失落、有满足、当然也有满满当当的收获。也算是不虚度这四年的光阴，这一切都离不开在这所校园内外为我提供帮助的各位老师、同学们的帮助。值此毕业论文完成之际，我在此谨向他们致以我最诚挚的谢意！

首先，感谢我毕业论文的校内指导老师——吴乐秦老师，吴乐秦老师不仅为我的毕业论文提供了重要的指导，而且在四年的学习中，吴老师独特新颖的思考方式给我的数学学习中带来了许多的启发，也感谢吴老师给予我助担任教的锻炼机会，感谢他为我所做的一切！

感谢我毕业论文的校外指导老师——马锦华老师，马老师是我学术的引路人，在我推免结束后带我进入实验室，给予了我许多学术指导，让我更加坚定地走上科研的道路，并且在毕业论文中为我提供了许多宝贵的建议和意见。

我还要感谢我们的数学系主任——杜毅老师，杜老师对我的帮助不仅仅是在数学的学习上，同时在生活以及人生方向上给予了我许多帮助。正是因为杜老师在大一期间的劝诫以及在我参加推免时的沟通，让我对我的个人发展有了进一步的认识。感谢杜老师对我学习和生活的关心与鼓励。

最后，感谢我大学四年所有的任课老师，感谢陈翔师兄，段虹宇师姐，温海贤和欧阳伟昊同学对我在学习科研以及推免上的帮助。感谢辅导员江秀海老师的鼓励与生活上的帮助。感谢母校暨南大学，是母校将我塑造为更加优秀的人，祝母校越办越好，早日跻身世界一流大学！

参考文献

[1] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable Convolutional Networks[C]. In The IEEE International Conference on Computer Vision (ICCV), pages 764-773, 2017. 1, 2, 5, 6, 7, 8

[2] Kaiming He, Georgia Gkioxari, Piotr Dollar, & Ross Girshick. (2017). Mask R-CNN[C]. In 2017 IEEE International Conference on Computer Vision (ICCV).

[3] Gao Huang, Zhuang Liu, L v. d. Maaten, and Kilian Q Weimberge. Densely Connected Convolutional Networks[C]. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2261-2269, 2017. 1

[4] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks[C]. In Neural Information Processing Systems (NIPS), pages 1-9, 2012. 1

[5] Karen Simonyan and Andrew Zisserman. Very Deep Convolution Networks for Large-Scale Image Recognition[C]. CoRR arXiv:1409.1556, abs/1409.1:1-14,2014. 1

[6] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V. Le. Learning Transferable Architectures for Scalabel Image Recognition[C]. arXiv:1707.07012,2017. 1

[7] Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. Few-shot object detection via feature reweighting[C]. In The IEEE International Conference on Computer Vision (ICCV), 2019. 1, 2, 4, 6, 7, 8

[8] Leonid Karlinsky, Joseph Shtok, Sivan Harary, Eli Schwartz, Amit Aides, Rogerio Feris, Raja Giryes, and Alex M. Bronstein. Repmet: Representative-based metric learning for classification and few-shot object detection[C]. In 2019 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019. 2

[9] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks[C]. In International Conference on Machine Learning (ICML), 2017. 2, 6, 7

[10]Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms[C]. arXiv, 2018. 2, 6, 8

[11]Ross Girshick, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]. In 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014. 2, 3

[12]He Kaiming, Zhang Xiangyu, Ren Shaoqing, Sun Jian. Spatial pyramid pooling in deep convolutional networks for visual recognition[C]. In European Conference on Computer Vision (ECCV). 2014.

[13]Ross Girshick. Fast R-CNN[C]. In The IEEE International Conference on Computer Vision (ICCV), 2015. 1, 2, 3, 4, 6

[14]Shaoqing Ren，Kaiming He，Ross Girshick, and Sun Jian. Faster R-CNN: Towards real-time object detection with region proposal networks[C]. In Neural Information Processing Systems (NIPS), 2015. 1, 2, 3, 4, 7

[15]Koch, Gregory. Siamese neural networks for one-shot image recognition[C]. ICML Deep Learning Workshop, 2015..

[16]Ravi, Sachin and Larochelle, Hugo. Optimization as a model for few-shot learning[C]. In International Conference on Learning Representations (ICLR), 2017.

[17]Hochreiter S, Schmidhuber J. Long Short-Term Memory[J]. Neural Computation, 1997, 9(8):1735.

[18]Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature Pyramid Networks for Object Detection[C]. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017. 3,

[19]He, Kaiming and Zhang, Xiangyu and Ren, Shaoqing and Jian, Sun. Deep Residual Learning for Image Recognition[C]. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[20]Chen, Liang-Chieh, Papandreou, George, Kokkinos, Iasonas, et al. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs[J]. Computer ence, 2014(4):357-361.

[21]Yu, Fisher, Koltun, Vladlen. Multi-Scale Context Aggregation by Dilated Convolutions[C]. In International Conference on Learning Representations (ICLR). 2016.

[22]Vinyals, Oriol and Blundell, Charles and Lillicrap, Timothy and Kavukcuoglu, Koray and Wierstra, Daan. Matching Networks for One Shot Learning[C]. In Neural Information Processing Systems (NIPS), 2016.

[23]Joseph Redmon and Ali Farhadi. Yolo9000: Better, faster, stronger[C]. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 6517–6525. IEEE, 2017. 1, 2, 3, 5

[24]Joseph Redmon, Santosh Divvala, Ross Girshick, and AliFarhadi. You only look once: Unified, real-time object detection[C]. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 779–788, 2016. 2

[25]Everingham, Mark and Luc Van Gool and Christopher K. I. Williams and John Winn and Andrew Zisserman. The Pascal Visual Object Classes (VOC) Challenge[J]. International Journal of Computer Vision, 2010, 88(2):303-338.

[26]Santoro, Adam, Bartunov, Sergey, Botvinick, Matthew, Wierstra, Daan, and Lillicrap, Timothy. Meta-learning with memory-augmented neural networks[C]. In International Conference on Machine Learning (ICML), 2016

[27]Hao Chen, Yali Wang, Guoyou Wang, and Yu Qiao. Lstd: A low-shot transfer detector for object detection[C]. The Association for the Advancement of Artificial Intelligence (AAAI), 2018. 1, 3, 6